Real-Time Image Edge Pixels Detection and Various Segmentation Techniques in Target Recognition

R.Sudha[Ω1], K.Adlin Suji*[1], A.Mani[#2]

[Ω] *Assistant Professor, Department of Computer Science,*
T.R.P. Engineering College, Irungalur, Trichy, Tamilnadu, India-621105
miralous@gmail.com
[*] *Associate Professor, Department of Computer Applications,*
Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamilnadu, India- 621212
Suji.adlin@gmail.com
[#] *Assistant Professor, Department of Computer Applications,*
Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamilnadu, India- 621212
inovic.mani@gmail.com

***Abstract*— This paper describes techniques to perform efficient and accurate target recognition in different domains. In order to accurately model small, irregularly shaped targets, the target objects and images are represented by their edge maps, with a local orientation associated with each edge pixel and an overview of the methodologies and algorithms for segmenting 2D images as a means in detecting target objects embedded in visual images for Automatic Target Detection/Recognition applications. IT can be defined as a simple comparison between an observed image (image to recognize) and a reference image;**

***Keywords*— Target Detection, Image Processing, Image edge pixels, Automatic Recognition**

I. INTRODUCTION

This paper considers methods to perform automatic target recognition by representing target models and images as sets of oriented edge pixels and performing matching in this domain. Automatic Target Detection/Recognition (ATDR) is an application of pattern recognition for image processing that detects and identifies types of target objects. ATDR is a major objective in processing digital images for detecting, classifying, and tracking target objects embedded in an image. Various methods exist for detecting objects of known type in a particular environment or image. The traditional approach to automatic target detection/recognition is to convert the signal from the sensor into a digital image for further processing. Next step is to separate the target from its background or surrounding area by extracting a coarse shape or outline of the target object and then to identify the object from features describing the target object.

II. TARGET POSITION DETECTION

Given an image or scene, S, and a target image, P, the neural system is to find the coordinates of the target image, P, in the scene, S. Additionally, given any sub-scene S' containing the Target, P, the system is expected to find the image P and to return its coordinates with respect to S'. It is assumed that a reference scene S0 and a target image, P0, are known *a priori*. It is also assumed that the range, azimuth and elevation of the observation point of scene, S, from the target, P, are known to a reasonable accuracy. This information will be used to transform any sub-scene S' to the same scale as S0, so as to produce a scene S for analysis of where the target P is located in S. The target image is typically derived from a photograph and the image to be matched may come from a video camera or another photograph.

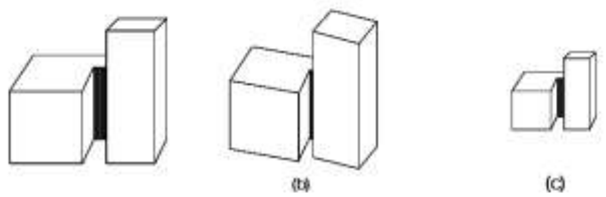


Figure 1. (a) The target image, (b) change of viewpoint, (c) change of scale

In the simple images in Fig. 1, if (a) is the target image, P0, then (b) and (c) are the same target seen from a different angle or range. It can be seen that (b) and (c) can be made to approximate (a) by the application of suitable geometric transforms. It is not possible to obtain a perfect reconstruction of (a) from (b) owing to the three-dimensional nature of the Image since these images are available to the system in two-dimensional form.

III. ATDR THROUGH SEGMENTATION

One of the common problems encountered in object detection/recognition is choosing a suitable approach for isolating different objects from each other as well as from the background. This separation of an image into object/s and background is usually done by simplifying and/or changing him representation of an image by enhancing the visual representation of boundaries (lines, curves, etc.). This makes the object differentiation, isolation and detection task easier. The process is known as image segmentation. Image segmentation is one of the primary steps in image analysis for object Detection, recognition and identification.

IV. MATCHING EDGE PIXELS IN ORDINARY IMAGE

This section first reviews the definition of the Hausdorff measure and how a generalization of this measure can be used to decide which object model positions are good matches to an image. This generalization of the Hausdorff measure yields a method for comparing edge maps that is robust to object occlusion, image noise, and clutter.

A. The Hausdorff Measure

The directed Hausdorff measure from M to I, where M and I are point sets, is

$$h(M, I) = \max_{m \in M} \min_{i \in I} \|m - i\|$$

where $\|\|$ is any norm. This yields the maximum distance of a point in set M from its nearest point in set I. In the context of recognition, the Hausdorff measure is used to determine the quality of a match between an object model and an image. If M is the set of (transformed) object model pixels and I is the set of image edge pixels, the directed Hausdorff measure determines the distance of the worst matching object pixel to its closest image pixel. Of course, due to occlusion, it cannot be assumed that each object pixel appears in the image. The partial Hausdorff measure [11] between these sets is thus often used. It is given by

$$h_K(M, I) = K^{th}_{m \in M} \min_{i \in I} \|m - i\|. \qquad (1)$$

This determines the Hausdorff measure among the K object pixels that are closest to image pixels. K can be set to the minimum number of object pixels that are expected to be found in the image if the object model is present or K can be set such that the probability of a false alarm occurring is small. Since this measure does not require that all of the pixels in the object model match the image closely, it is robust to partial occlusion. Furthermore, noise can be withstood by accepting models for which this measure is nonzero, and this measure is robust to clutter that may appear in the image since it measures only the quality of the match from the model to the image and not vice versa.

B. The Generalization to Oriented Points

The Hausdorff measure can be generalized to incorporate oriented pixels by considering each edge pixel in both the object model and the image to be a vector in $\mathbb{R}^3$:

$$\hat{p} = \begin{bmatrix} p_x \\ p_y \\ p_o \end{bmatrix}$$

Where (p_x, p_y) is the location of the point, and p_o is the local orientation of the point (e.g., the direction of the gradient, edge normal, or tangent). Typically, we are concerned with edge points on a pixel grid, and the x and y values thus fall into discrete sets. The orientations can be mapped into a discrete set in a similar manner. Let us call a set of image points that have been extended in this fashion an *oriented image edge map* $\hat{I}$, and similarly, let us call such an extended set of points in the object model an *oriented model edge map* $\hat{M}$.

We now need a measure to determine how well these oriented edge maps match. Among pixels with the same orientation, we would like the measure to reduce to the previous Hausdorff measure. Furthermore, the previous measure should be a lower bound on the new measure. One measure that fulfills these conditions is

$$h_a(\hat{M}, \hat{I}) = \max_{\hat{m} \in \hat{M}} \min_{\hat{i} \in \hat{I}} \max \left\{ \left\| \begin{bmatrix} m_x - i_x \\ m_y - i_y \end{bmatrix} \right\|, \frac{|m_o - i_o|}{a} \right\}.$$

Our system discretizes the orientations such that $a = 1$ and uses the L_∞ norm. In this case, the measure for oriented points simplifies to

$$h(\hat{M}, \hat{I}) = \max_{\hat{m} \in \hat{M}} \min_{\hat{i} \in \hat{I}} \|\hat{m} - \hat{i}\|_\infty .$$

V. EDGE PIXELS IN UNDERWATER IMAGE

In underwater imaging, visibility is a major problem. It varies from about 30 m in very clear water to about 0.75 m in very turbid water like a harbor water. The received image IR is the addition of the attenuated image IA, the forward scattered image IFS and the backscattered component

IBS:3 IR = IA + IFS + IBS (1)

The light attenuation increases exponentially with distance (according to the Beer-Lambert law) and limits visibility. The backscattered component corresponds to the light that has been reflected by particles towards the video camera (see Figure 1A). This hides objects present in the scene. The forward scattered component is the light reflected by the object that has been diffused by particles on its way back to the camera, which produces blurred images (see Figure 1A).

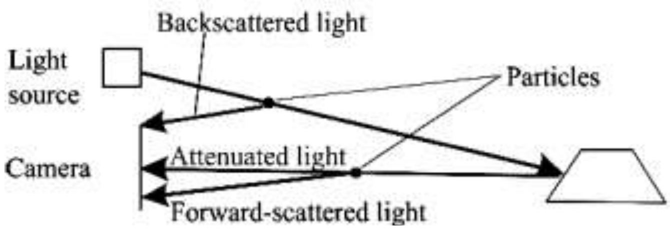


Figure 1.A. Diagram of the light propagation in an underwater medium

It is possible to add artificial lighting in the scene; this increases the visibility range but it produces a non-uniform lighting. Video detects also particles on which light is reflected, called marine snow. These consequences of the underwater medium are annoying; especially backscattering that is the most important effect. Thus underwater images need preprocessing.

A. *Studied targets*

The studied targets are underwater mines (Figure 2). There are many kinds of mines. In this work we focus on three of them: spherical mines (Figure 2-a), cylindrical mines (Figure 2-b) and Manta mines (Figure 2-c) (the Manta mine is a truncated cone).

Figure 2. Grayscale images representing three studied targets that come from underwater videos: (a) a spherical mine, (b) a cylindrical mine (c) and a Manta mine

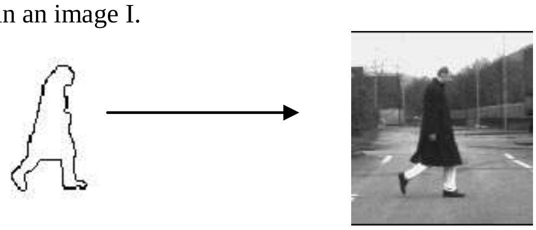
VI. SEARCH STRATEGY

Recent work has shown that efficient methods can be formulated to search the space of possible transformations of the model to find the position with the minimum Hausdorff measure or all positions where the measure is below some threshold. This section discusses how such methods operate in general and how they can be extended to oriented points. In addition, we describe techniques that are used to reduce the running time of the system when there are multiple object models that may appear in the image.

A. Matching Edge based Pixels in ordinary image

It is an edge matching technique that minimizes the sum of the distances from each object edge pixel to its closest image edge pixel over the space of possible transformations. This technique is closely related to minimizing the generalized Hausdorff measure, which instead minimizes the Kth largest of these distances. Since the cham-fer measure sums the distances over all of the object pixels, it is not robust to occlusion. In the original formulation of chamfer matching, Barrow *et al.* [1] used a starting hypothesis and an optimization procedure to determine a position of the model that is a local minimum with respect to the chamfer measure. This method requires a good starting hypothesis to converge to the global minimum.

We want to determine the presence and location of a template T in an image I.



Edge-Template **Image Scene**

Figure 3. Edge based Target Recognition

B. Using Oriented Pixels in ordinary image

Since the oriented object and image pixels have three degrees of freedom, a 3-D distance transform is now required. Before this can be computed, we must consider how rotations of object models will be treated since such rotations change the orientations of the object pixels. If we wish to rule out nearby transformations that may change the orientations of object pixels, then this must be accounted for the distance transform, but this is problematic since the discretization of the rotations in the transformation space will, in general, be very different from the discretization of the orientations of the edge pixels. To avoid this problem, each rotation of an object model is treated independently

It must also be decided how the models will be rotated and scaled to compare them to the image. If a CAD model is available from which the edges of our targets can be determined, these models can be rotated before performing the edge detection stage since different rotations of the model are treated as (essentially) separate models. On the other hand, if the original model consists only of a set of edge points, each point is simply rotated around the center of the model. Similarly, scaling of the model is performed by scaling each point with respect to the center of the model.

It is now possible to use Hausdorff matching techniques similar to those for unoriented points to perform efficient recognition.

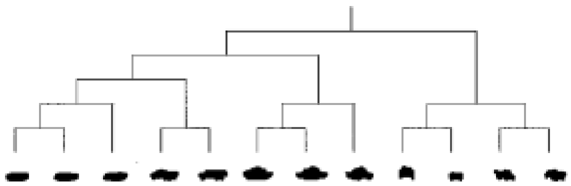


Figure 4. Hierarchical clustering of the models is performed as the canonical positions of the models relative to each other are determined. This figure s hows an example of the hierarchy produced by these techniques for 12 model views. The full silhouettes are shown rather than the edge maps for visual purposes.

To process a single cell, the following steps are performed. First, a discrete transformation close to the center of the cell is chosen, and the maximum difference in the transformed location of a model pixel between the center transformation and any other transformation in the cell must be computed. This is bounded by the sum of the distance in the scale direction (by counting the number of discrete scales) between the transformations and the maximum of the distances in the x and y directions since we use the L norm in the image space.

C. Considering Multiple Models in ordinary image

When there are multiple object models that may appear in a single image, there are methods by which the search can be made faster than examining each object model sequentially. This section describes one such method. Note that these object models need not come from separate objects; they may be alternate views of the same object.

The method builds a tree of models using hierarchical clustering techniques [8]. At each step, the two closest models are determined and clustered. This yields a canonical position for these models with respect to each other and a new set of model points replacing the two previous models. The new ªmodelº is then compared with the remaining models as above, and the process is repeated until all of the models belong to a single hierarchically constructed model tree. At this point, canonical positions for each model with respect to the others have been computed, and a model hierarchy represented by a binary tree has been determined, where the leaves of the tree are individual models, and the remaining nodes correspond to the set of models below them in the tree. Fig. 4 shows a small example.

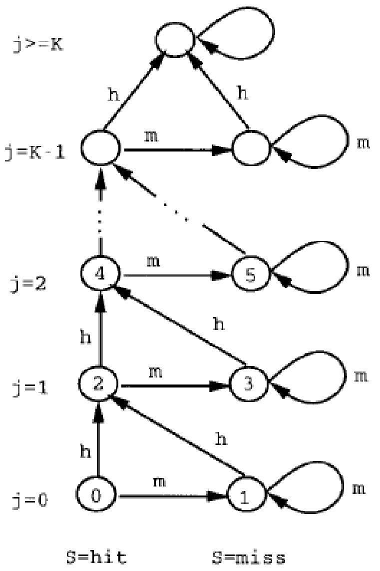


Fig. 5. Markov chain that counts the number of object pixels that match image pixels.

D. Correlation for water image

Correlation is a signal and image processing method that compares a target image with a reference image. The result of this comparison is a more or less intense correlation peak, depending on the resemblance degree between these two images. Mathematically, correlation can be written:

c(x0, y0) = h(x0, y0) _c s(x0, y0)

$$c(x_0, y_0) = \int \int_{-\infty}^{+\infty} h^*(x, y) s(x + x_0, y + y_0) dx dy$$

where "c" is the result of the correlation operation, "∗c"is the correlation product,"∗" is the complex conjugate operator, "h" is a filter or reference, "s" is the image to analyze, "(x0, y0)" are spatial coordinates and "(x,y)" are integration variables.

The correlation operation can be expressed with a Fourier transform:

C(μ, v) = H∗(μ, v).S(μ,v)

where μ and v are the coordinates in the frequency plane, C, S and H∗ are the respective Fourier transforms of functions c, s and h∗. The classical matched filter has been modified by introducing information in order to obtain some robustness to noise or orientation change for instance. Here we develop only three filters: classical matched filter, phase only filter (POF) and optimal trade-off filter (OT filter). Other filters have been developed, in the literature, like the binary phase only filter and the inverse filter. These filters, expressed in the Fourier plane, may then replace "H∗" in equation.

The most known filter is the classical matched filter (F_{CMF}) defined in the Fourier plane as:

F_{CMF} (μ, v) = ∞S∗(μ, v) / B(μ, v)

where "S∗(μ,v)" denotes the complex conjugate of the reference, "B" the spectral density of the background and "∞" is a constant. This filter is robust but has a low discriminating power. The phase only filter (F_{POF}) is defined in the Fourier plane as:

F_{POF} (μ, v) =S∗(μ,v) / |S∗(μ,v)|

Where "|S∗(μ, v)|" is the module of the reference spectrum. This filter gives a sharp correlation peak, it is very discriminative but also noise sensitive. The optimal trade-off filter (F_{OT}) is defined in the Fourier plane as

F_{OT} (μ,v) =S∗(μ, v) / ∞B(μ, v) + (1 − ∞)|S∗(μ, v)|

From now on, in this article, we will use only the POF filter. Since we need a discriminative filter. We don't use the OT filter because in order to obtain good performances we have to know the spectral density of the background which must be very close to the one in the actual images. In underwater applications, it is very difficult to know the spectral density of the background because of the ocean floor, turbidity and all noises that perturb an image and vary throughout the video.

Correlation is extremely fast. The fast Fourier transform takes O(n log(n)) arithmetical operations and each image multiplication takes O(n2) arithmetical operations, where n is one of the two size of the image to analyze (here, we suppose that image is a square image).

VII. PROBABILITY OF A FALSE ALARM

A. A Simple Model for Matching Oriented Pixels

Let us consider matching a single connected chain of oriented object pixels to the image at some specified location. then the process is said to be a *Markov process*. If, furthermore, the probability does not depend on i, then the process is a *Markov chain*. To determine the probability distribution of the number of hits over the entire object model, the number of hits so far in our chain j must be counted explicitly. A separate state in the chain is thus used for each member of

$$\{hit, miss\} \times \{j \mid 0 \leq j \leq m\}$$

where m is the number of object pixels. If we are only interested in whether a false alarm of size K occurs, a Markov chain with $2K+1$ states can be used .If the final state of this chain is reached due to matches with random edge chains in the image, then a false alarm has occurred.

Let p_0 be a vector containing the probability of the chain starting in each state. The probability distribution among the states after examining the entire object chain is

$$p_m = T^m p_0.$$

The last element of p_m is the probability that a false alarm of size K will occur at this position of the model. The probability that a false alarm of any other size $K_f \leq K$ will occur can be determined by summing the appropriate elements of p .

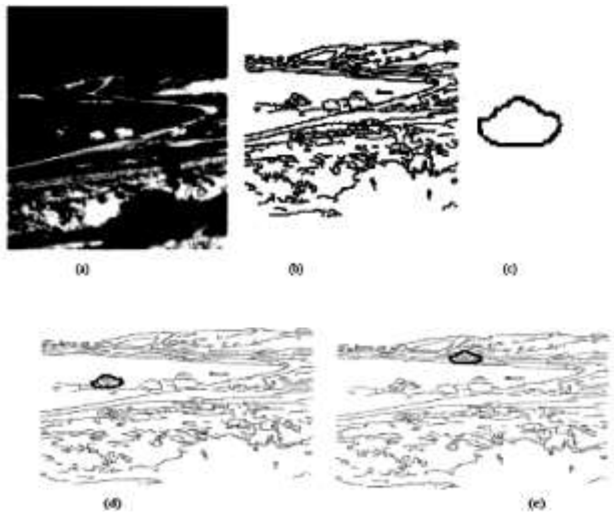


Figure. 6. Automatic target recognition example. (a) FLIR image after histogram equalization. (b) Edges found in the image. (c) Smoothed edges of a tank model. (d) Detected position of the tank. (e) False alarm.

B. An Accurate Model for Matching

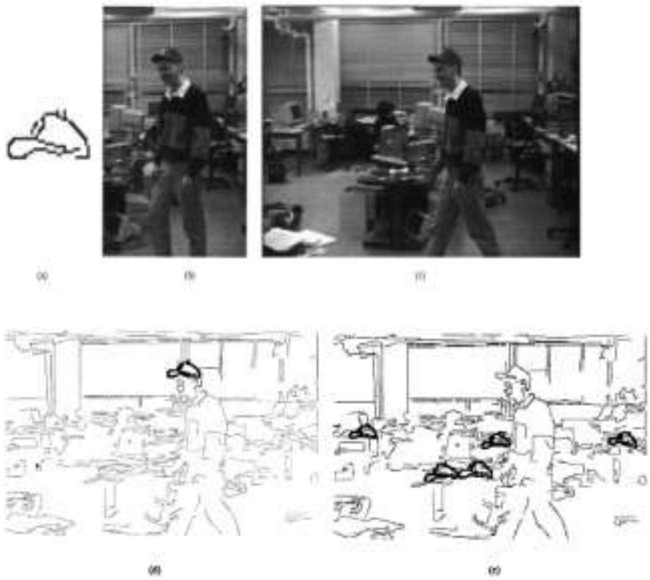
To model the matching process accurately, it is not correct to treat the state transition probabilities as independent of which pixel in the chain is examined.

Its means that the stochastic process of pixel hits and misses is not a Markov chain, but it is still a Markov process. Let T be the state transition matrix for the ith object pixel in such a process. The state probability vector p is now given by

$$p_m = \left(\prod_{i=0}^{m-1} T_i\right) p_0.$$

- m: The object pixel did not hit an image pixel.
- n: The object pixel hit a new pixel in the oriented image edge map.
- o: The object pixel hit the same pixel in the oriented image edge map as the previous object pixel.
- p: The object pixel hit the same pixel in the oriented image edge map as the previous two object pixels.

$$\{m, n, o, p\} \times \{j \mid 0 \leq j \leq K\}.$$



Figugre. 7. Image sequence example. (a) Object model. (b) Part of the image frame from which the model was extracted. (c) Image frame in which we are searching for the model. (d) Position of the model located using orientation information. No false alarms were found for this case. (e) Several false alarms that were found when orientation information was not used. These each yielded a higher score than the correct position of the model.

C. State Transition Probabilities

The state transition probabilities must now be determined. These probabilities will be different in locations of the image that have different densities of edge pixels. Consider, for example, the probability of hitting a new pixel following a miss. The probability will be much higher if the window is dense with edge pixels rather than having few edge pixels. To model this, let us consider the window of the image that the object model overlays at some position. This is simply the rectangular sub image covered by the object model at this position. Each of these windows in the image will enclose some number d of image pixels. We call this the density of the image window. The state transition probabilities are closely approximated by linear functions of the number of edge pixels present in the image window and belong to one of two classes:

1) Probabilities that are linear functions passing through the origin (i.e., Pr $= k_i d$): The probability that an object model pixel hits a new image pixel, when the previous object model pixel did not hit a new pixel, is approximated by such a linear function of the density of image edge pixels in the image window. The following state transition probabilities are thus modeled in this manner: $P_{nm}(i)$, $P_{sm}(i)$ and $P_{pm}(i)$. Note that each has a different constant k_i.
2) Probabilities that are constant (i.e., Pr $= c$): When the previous object model pixel hit an image pixel, the probability that the current object model pixel will hit the same image pixel is essentially constant. In addition, when the object model chain is following an image chain (i.e., the previous object model pixel hit a new image pixel), the probability that the object model chain continues to follow the image chain is approximately constant. The state transitions that are modeled in this manner are thus $k_i \cdot P(i) P_A(i)$, and $P(i)$.

The remaining probabilities can be determined as a function of these probabilities as follows:

$$P_{nn}(i) = 1 - P_{ns}(i) - P_{nm}(i)$$
$$P_{ss}(i) = 1 - P_{sm}(i) - P_{sp}(i)$$
$$P_{pp}(i) = 1 - P_{pm}(i)$$
$$P_{mm}(i) = 1 - P_{mn}(i).$$

If the state at $i=0$ is considered to be m, this will yield the correct result for the first pixel in the object chain (i.e., $i=1$). In this case, there are no previous object model pixels to compare against, and the probability of an object pixel resulting in a hit at random is desired. Similarly, if the object model consists of more than one chain of pixels, the state is reset to m when a new chain is started.

D. Probability of a False Alarm Over a Set of Transformations

Let us now consider the probability that there exists a false alarm at any translation of the object model. As with the search strategy, only translations on the integer grid are considered. While this may miss the optimal translation for our matching measure, this can increase the size of the minimum Hausdorff measure over the space of possible translations by at most 1 when using the L_2 norm.

We do not assume that a target model will always appear either brighter or darker than the background in an image, but we do assume that individual targets will be either entirely brighter or entirely darker than the background, although this restriction can be easily removed. This means that each translation must be considered twice: once for the case when the target is brighter than the background and once for the case when the target is darker since the orientation of the point in these two cases will be shifted by π. If $P_K(t)$ is the probability of a false alarm of size K at translation t, the probability of a false alarm existing over all translations can be determined by computing

$$1 - \prod_t (1 - P_K(t)).$$

This can be computed more efficiently if we have a histogram of the number of edge pixels contained in the image windows. Let d_i be the number of image windows containing i edge pixels for $0 \le i \le W$, where W is the size of the window in pixels. The probability of a false alarm in two image windows containing the same number of image pixels is the same in this estimation model. Let $P_K(i)$ be the probability of a false alarm of size K in a window containing i edge pixels. The probability of a false alarm is now given by

$$1 - \prod_{i=0}^{W} (1 - P_K(i))^{d_i}. \quad (3)$$

E. Using the False Alarm Rate Estimate

Now that we have a method to estimate the probability of a false alarm for any particular matching threshold, we can use the estimate to improve the performance of a recognition system that matches oriented edge pixels.

One method by which we could use the estimate is to set the matching threshold such that the probability of a false alarm is below some predetermined probability. However, this can be problematic in very cluttered images since it can cause correct instances of targets that are sought to be missed.

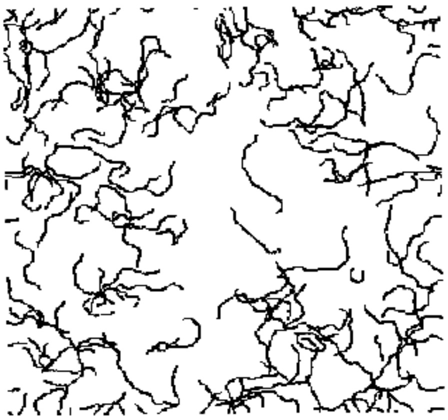


Fig. 8. One of the synthetic images used to generate ROC curves.

VIII. PERFORMANCE

Fig. 4 shows an example of the use of these techniques. The image is a low contrast infrared image of an outdoor terrain scene. After histogram equalization, a tank can be seen in the left-center of the image, although due to the low contrast, the edges of the tank are not clearly detected.

The current implementation of these techniques uses 16 discrete orientations and $\delta - \sigma - 1$ (each discrete orientation thus corresponds to π rad, but matches are also allowed with neighboring orientations). In these experiments, the allowable orientation and scale change of the object views was limited to $+\pi$ and $+10\%$, respectively, since we expect to have prior knowledge of the approximate range and orientation of the target.

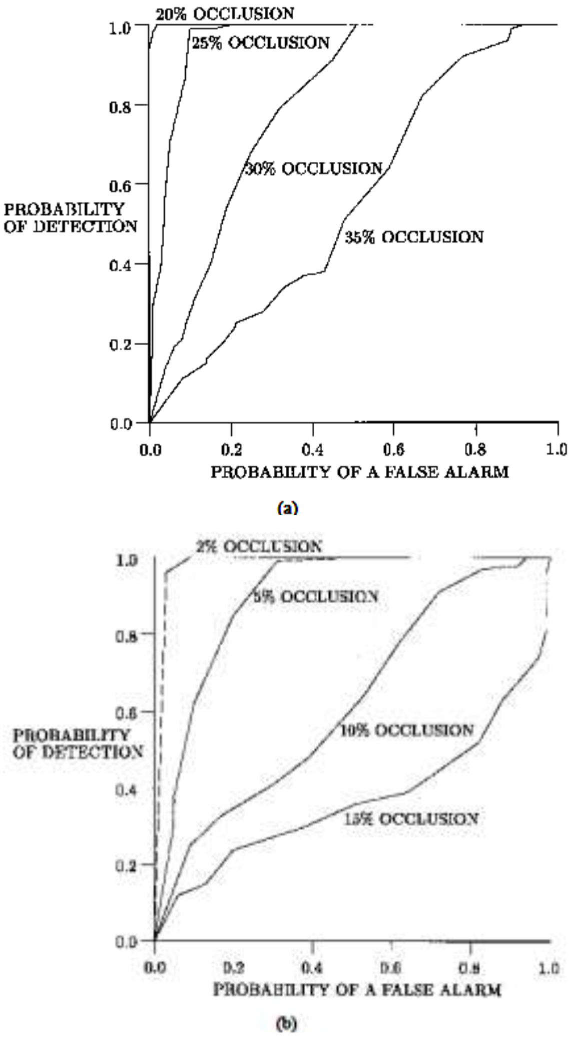


Fig. 9. Receiver operating characteristic (ROC) curves generated using synthetic data. (a) ROC curves when using orientation information. (b) ROC curves when not using orientation information.

These techniques are not limited to automatic target recognition. Fig. 6 shows an example of the use of these techniques in a complex indoor scene. In this case, the object model was extracted from a frame in an image sequence, and it is matched to a later frame in the sequence (as in tracking applications). Since little time has passed between these frames, it is assumed that the model has not undergone much rotation out of the im-age plane, and thus, a four-dimensional (4-D) transformation space is used, consisting of translation, rotation in the plane, and scale. The position of the object was correctly located when orientation information was used. No false alarms were found for this case. When orientation information was not used, several positions of the object were found that yielded a better score than the correct position of the object.

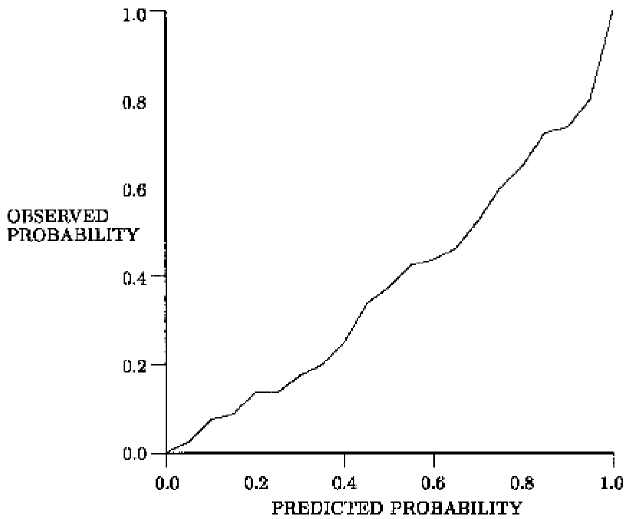


Fig. 10. Predicted probability of a false alarm versus observed probability of a false alarm in trials using real images.

We have generated ROC curves for this system using synthetic edge images. Each synthetic edge image was generated with 10% of the pixels filled with random image clutter (curved chains of connected pixels). An instance of a target was placed in each image with varying levels of occlusion generated by removing a connected segment of the target boundary. Random Gaussian noise was added to the locations of the pixels corresponding to the target. The false alarm rate (FAR) estimation techniques were tested on real imagery. In these tests, the largest threshold at which a false alarm was found was determined for each object model and image in a test set. In addition, the FAR estimation techniques were used to determine the probability that a false alarm of at least this size would be determined in each case.

IX. SEGMENTATION TECHNIQUES

Several general-purpose techniques and algorithms have been developed for image segmentation. Since there is no general solution to the image segmentation problem, these techniques often have to be combined with domain knowledge in order to Effectively solve an image segmentation problem for a problem domain. Thus Image segmentation needs to be approached from a wide variety of perspectives. The following approaches of image segmentation are reviewed in this paper.

a) Edge Detection Methods
b) Histogram Based Methods
c) Tree/Graph Based Methods
d) Region Splitting Methods
e) Region Growing Methods
f) Model Based segmentation
g) Neural Network Based segmentation
h) Clustering Methods
i) Graph Partitioning Methods
j) Watershed Transformation
k) Multiscale segmentation
l) Probablistic and Bayesian approaches

a). Edge-Detection Methods

Edge detection is a well-developed field on its own within image processing. Region boundaries and edges are closely related, since there is often a sharp adjustment in intensity at the region boundaries. Edge detection techniques have therefore been used as the base of another segmentation technique.

b). Histogram-Based Methods

In this technique, a histogram is computed from all of the pixels in the image, and the peaks and valleys in the histogram are used to locate the clusters in the image. Color or intensity can be used as the measure

c). Tree/Graph Based Methods

Proposed a new approach for segmentation, which is derived from the consensus of a set of different segmentation outputs on one input image. Instead of statistics characterising the spatial structure of the local neighborhood of a pixel, for every pair of adjacent pixels the collected statistics are used for determining local homogeneity. From the ensemble of these initial segmentations, for every adjacent pixel pair a co-occurrence probability is derived, which captures global information (about the image) at the local level (pixel level). The final segmentation of the input image is obtained by processing the co-occurrence probability field.

d) Region Splitting Methods

This is ***divide and conquer*** or ***top down*** method. In this method, the image is split or broken into a set of disjoint regions which are similar within themselves

- Initially the entire image is treated as area of interest.
- Next identify a region within the image which satisfy some similarity constraint.
- If **TRUE** then the area of interest corresponds to a region in the image.
- If **FALSE** split the region of interest and consider each of the sub-regions as the region of interest in turn.
- This process continues until no further splitting occurs.
- In the worst case this happens when the areas are just one pixel in size.

e) *Region Growing Methods*

Region growing is the opposite of the split and merges approach, Thus it is a **bottom-up** Approach. The steps are as below

- An initial set of small areas are identified and iteratively merged according to similarity constraints.
- An arbitrary ***seed pixel*** is chooses to begin with and is compared with neighbouring pixels.
- Similar neighbouring pixels are added to the seed pixel and the region is ***grown*** by increasing the size of the region.
- When no more similar pixels are available, another pixel which does not yet belong to any region is selected and the process is started again.
- The process is continued until all pixels belong to some region.

f) *Model based Segmentation*

The central assumption of Model Based approach is that structures of interest/objects have a repetitive form of geometry. Therefore, one can seek for a probabilistic model towards explaining the variation of the shape of the object and then when segmenting an image impose constraints using this model as prior. Such a task involves

(i) Registration of the training examples to a common pose,
(ii) Probabilistic representation of the variation of the registered samples, and
(iii) Statistical inference between the model and the image. State of the art methods in the literature for knowledge-based segmentation involve active shape and appearance models, active contours and deformable templates and level-set based methods.

g). *ANN Based segmentation*

Artificial Neural Network segmentation relies on processing small areas of an image using a neural network or a set of neural networks. After such processing the decision-making mechanism marks the areas of an image accordingly to the category recognized by the neural network.

h). *Clustering Method segmentation*

Image segmentation can be performed effectively by clustering image pixels. Cluster analysis allows the partitioning of data into meaningful subgroups and it can be applied for image segmentation or classification purposes.

Clustering analysis either requires the user to provide the seeds for the regions to be segmented or uses non-parametric methods for finding the salient regions without the need for seed points. Clustering is commonly used for image segmentation and unsupervised learning.
The **k-means algorithm** is an algorithm to cluster n objects based on attributes into k partitions or groups, $k < n$. The K-means algorithm is an iterative technique that is used to partition an image into K clusters. The basic algorithm is:

- Pick K cluster centers, either randomly or based on some heuristic
- Assign each pixel in the image to the cluster that minimizes the variance between the pixel and the cluster center.
- Re-compute the cluster centers by averaging all of the pixels in the cluster.
- Repeat steps 2 and 3 until convergence is attained (e.g. no pixels change clusters)

i) *Graph Partitioning Method*

In this method, the image being segmented is modelled as a weighted undirected graph. Each pixel is a node in the graph, and an edge is formed between every pair of pixels. The weight of an edge is a measure of the similarity between the pixels. The image is partitioned into disjoint sets (segments) by removing the edges connecting the segments. The optimal partitioning of the graph is the one that minimizes the weights of the edges that were removed (the "cut"). **Shi's** algorithm seeks to minimize the "normalized cut", which is the ratio of the "cut" to all of the edges in the set.

j) *Watershed Transformations*

The Watershed transformation considers the gradient magnitude of an image as a topographic surface. Pixels having the highest gradient magnitude intensities (GMIs) correspond to watershed lines, which represent the region boundaries. Water placed on any pixel enclosed by a common watershed line flows downhill to a common local intensity minima (LMI). Pixels draining to a common minimum form a catchment basin, which represent the regions

k) *Multiscale segmentation*

Multi-scale segmentation is a general framework for signal and image segmentation, based on the computation of image descriptors at multiple scales of smoothing. It included the notion that a one dimensional signal could be unambiguously segmented into regions, with one scale parameter controlling the scale of segmentation. When studied the problem of linking local extrema and saddle points over scales, and proposed an image representation called the *scale space primal sketch* which makes explicit the relations between structures at different scales, and also makes explicit which image features are stable over large ranges of scale including locally appropriate scales for those. Proposed to detect edges at coarse scales in scale space and then trace them back to finer scales with manual choice of both the coarse detection scale and the fine localization scale.

l) *Probabilistic and Bayesian approaches*

Use co-occurrence based approach to image segmentation making use of region and boundary information in parallel for improved performance on a sequence of images. In this method, initial segmentation is done based on the location of the intensities of each pixel and its neighbours in the co-occurrence matrix. Each pixel is then associated with a tuple which specifies whether it belongs to a given region or if it is a boundary pixel. This tentative segmentation was then refined. The algorithm is less effective if the clusters in the co-occurrence space have substantial overlap due to the imposition of local consistency. Since the techniques use global information in a local context, it was possible to adapt it to varying image characteristics i.e. variation in colour and texture.

X. CONCLUSION

This paper has discussed techniques to perform automatic target recognition by matching sets of oriented edge pixel in ordinary images, underwater images and importance of segmentation in image analysis and various methods of segmentation.

We also discussed about matching technique of both ordinary image and under water image, a search strategy that allowed the full space of possible transformations to be examined quickly in practice using a hierarchical cell decomposition of the transformation space was then given. This method allows large volumes of the transformation space to be efficiently eliminated from consideration.

Additional techniques for reducing the overall time necessary when any of several target models may appear in an image were also described. The probability that this method would yield false alarms due to random chains of edge pixels in the image was discussed in detail, and a method to estimate the probability of a false alarm efficiently at run time was given. This allows automatic target recognition to be performed adaptively by maintaining the false alarm rate at a specified value or to rank the competing hypotheses that are found on their likelihood of being a false alarm.

Experiments confirmed that the use of orientation information at each edge pixel, in addition to the pixel locations, considerably reduces the size and number of false alarms found. The experiments also indicated that the use of orientation information resulted in faster recognition.

And finally, we discussed almost all image segmentation techniques proposed so far are *ad hoc* in nature. There are no general algorithms that will work for all images. One of the main objectives of segmentation algorithm is to precisely segment the image without under or over segmentation.

XI. REFERENCES

[1] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, ªParametric correspondence and chamfer matching: Two new techniques for image matching,º in *Proc. Int. Joint Conf. Artificial Intell.* , 1977, pp. 659±663.

[2] B. Bhanu, ªAutomatic target recognition: State of the art survey,º *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-22, pp. 364±379, July 1986.

[3] B. Bhanu and T. L. Jones, ªImage understanding research for automatic target recognition,º *IEEE Aerospace Electron. Syst. Mag.*, vol. 8, pp. 15±23, Oct. 1993.

[4] G. Borgefors, ªDistance transformations in digital images,º *Comput. Vision, Graphics, Image Processing*, vol. 34, pp. 344±371, 1986.

[5] ——, ªHierarchical chamfer matching: A parametric edge matching algorithm,º *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 10, pp. 849±865, Nov. 1988.

[6] W. M. Brown and C. W. Swonger, ªA prospectus for automatic target recognition,º *IEEE Trans. Aerospace Electron. Syst.*, vol. 25, no. 3, pp. 401±410, May 1989.

[7] C. E. Daniell, D. H. Kemsley, W. P. Lincoln, W. A. Tackett, and G. A. Baraghimian, ªArtificial neural networks for automatic target recognition,º *Opt. Eng.*, vol. 31, no. 12, pp. 2521±2531, Dec. 1992.

[8] W. H. E. Day and H. Edelsbrunner, ªEfficient algorithms for agglomerative hierarchical clustering methods,º *J. Classification* , vol. 1, no. 1, pp. 7±24, 1984.

[9] D. E. Dudgeon and R. T. Lacoss, ªAn overview of automatic target recognition,º *Lincoln Lab. J.*, vol. 6, no. 1, pp. 3±9, 1993.

[10] W. E. L. Grimson and D. P. Huttenlocher, ªAnalyzing the probability of a false alarm for the Hausdorff distance under translation,º in *Proc. Workshop Performance versus Methodology Computer Vision*, 1994, pp. 199±205.

[11] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, ªComparing images using the Hausdorff distance,º *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, pp. 850±863, Sept. 1993.

[12] D. P. Huttenlocher and W. J. Rucklidge, ªA multi-resolution technique for comparing images using the Hausdorff distance,º in *Proc. IEEE Conf. Comput. Vision Patt. Recogn.*, 1993, pp. 705±706.

[13] C. F. Olson and D. P. Huttenlocher, ªRecognition by matching dense, oriented edge pixels,º in *Proc. Int. Symp. Comput. Vision*, 1995, pp. 91±96.

[14] ªDetermining the probability of a false positive when matching chains of oriented pixels,º in *Proc. ARPA Image Understanding Workshop*, 1996, pp. 1175±1180.

[15] C. F. Olson, D. P. Huttenlocher, and D. M. Doria, ªRecognition by matching with edge location and orientation,º in *Proc. ARPA Image Understanding Workshop*, 1996, pp. 1167±1174.

[16] D. W. Paglieroni, ªDistance transforms: Properties and machine vision applications,º *CVGIP: Graphical Models Image Processing*, vol. 54, no. 1, pp. 56±74, Jan. 1992.

[17] D. W. Paglieroni, G. E. Ford, and E. M. Tsujimoto, ªThe position-orientation masking approach to parametric search for template matching,º *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, pp. 740±747, July 1994.

[18] A. Rosenfeld and J. Pfaltz, ªSequential operations in digital picture processing,º *J. Assoc. Comput. Mach.*, vol. 13, pp. 471±494, 1966.

[19] W. J. Rucklidge, ªLocating objects using the Hausdorff distance,º in *Proc. Int. Conf. Comput. Vision*, 1995, pp. 457±464.

[20] F. Sadjadi, ªObject recognition using coding schemes,º *Opt. Eng.*, vol. 31, no. 12, pp. 2580±2583, Dec. 1992.

[21] J. G. Verly, R. L. Delanoy, and D. E. Dudgeon, ªModel-based system for automatic target recognition from forward-looking laser-radar imagery,º *Opt. Eng.*, vol. 31, no. 12, pp. 2540±2552, Dec. 1992.

[22] E. G. Zelnio, ªATR paradigm comparison with emphasis on model-based vision,º in *Proc. SPIE, Model-Based Vision Development Tools*, vol. 1609, 1992, pp. 2±15.

[23] Clark F. Olson and Daniel P. Huttenlocher Automatic Target Recognition by Matching Oriented Edge Pixels IEEE Transactions on image processing VOL. 6, NO. 1, JANUARY 1997

[24] G.N.Srinivasan, Dr. Shobha G. Segmentation Techniques for Target Recognition. International Journal Of Computers And Communications Issue 3, Volume 1, 2007.

[25] Thirumalini.S et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3) , 2014, 2882-2885.

[26] Isabelle Leonard, Andreas Arnold Bos, Ayman Alfalou. Interest of correlation-based automatic target recognition in underwater optical images: theoretical justi_cation and _rst results. SPIE Defense, Security, and Sensing, Apr 2010, Orlando, United States. 7678, pp.76780O, 2010.

[27] *Dr Trevor Clarkson* Automatic Target Recognition using Neural Networks. http://clarkson.me.uk/academic/atrpram.pdf.