URKUND

Urkund Analysis Result

Analysed Document: Thesis_RSV_object_Detection.docx (D20909668)
Submitted: 2016-06-20 13:38:00
Submitted By: suchitra.choudhary2007@gmail.com
Significance: 0 %

Sources included in the report:

Instances where selected sources appear:

0

CHAPTER 1 INTRODUCTION 1.1 Overview Detecting object in images is one of the major focuses of image processing and computer vision applications. Object Detection refers to the process of examining the images to determine the presence and location of the object. The task of detecting object in arbitrary environment is challenging for variety of reasons. The object could be present in different postures and sizes and also captured from different viewing angles. Apart from this, in realistic environment, the object appears in a cluttered background and under different illumination conditions. Objects are vital and significant part of our day to day life. Humans perceive objects through senses without any efforts. But machines do not possess the intelligence to perceive and recognize objects by itself. In order to develop the understanding of objects for the machines, the objects are to be described based on different properties. For the machine to detect the object, some prior knowledge about the object to be searched i.e. the model of the object, must be known in advance. The process of model creation involves the representation of object using different features, based on object properties. Different techniques use different properties of objects for representation like color, shape, intensity, texture etc [1]. Basic steps for object detection include feature extraction and feature matching with object template created using similar features. There exist certain challenges for detection of objects due to environmental conditions and object properties. For efficient object detection, the detection system should be invariant to these challenges. 1.2 Challenges in Object Detection (i) Illumination Variation: Images of the same object vary at different places due to variation in illumination conditions due to which the intensity of different pixels in the object image varies. Detection algorithm should be invariant to illumination changes (ii) Position Variation: Object position in the image can be changed. Object detection system should handle the variation in the position of the object in the image. (iii) Rotation Variation: Object can be placed at different orientation. The system must be able to detect the object at any orientation (iv) Scale Variation: Difference in the size of the object should not affect the detection of the object (v) Occlusion: Detection system should be able to detect the object from the image even if the object is not completely visible. An efficient detection system should possess invariance to all the above mentioned challenges. 1.3 Application Object detection has application in the varied fields. Automatic detection plays a vital role in image understanding and computer vision applications. With the enormous growth in the volume of digital information, manual handling and operation has become unmanageable. Continuous improvement in automatic object detection techniques has solved complex challenges in different fields of interest, like video surveillance, medical image analysis, defense applications, urban area extraction and object detection in aerial images [2]. Apart from these areas, object detection has proved to be very efficient in industrial applications for position measurement for example locating the electrical components on the printed circuit board, inspection i.e. quality control in production field, sorting, counting of the objects etc. It has its application in image retrieval and scene categorization [1]. 1.4 Object Detection Process Object Detection process involves extraction of features which describe the object; the extracted features are used to form the object hypotheses. The object model database contains the template objects which are basically the feature vectors generated from the object based on the properties of the object. Using the features extracted from the image the hypothesizer determines the presence and location of the object in the image. It matches the features extracted from the image to the features of object model from the database. The components of the object detection system are shown in fig 1.1. Fig 1.1 Object Detection system Object Features can be extracted based on the appearance and shape of the objects. There are various properties of the objects i.e. color, texture, pixel intensity, boundary points, dimensions etc which can be used to represent the object. Object can be represented by set of points which occupy space in the image, using primitive geometric shapes like rectangle, circle etc. Object representation can also be done using the contour points or the object silhouette. Other representation of the objects are based on articulated shape models, for example, human body is a articulated object with head, hands, legs connected by joints[15]. Most of the detection algorithms involve the classifier for the detection of object using a sliding window over the image where object is to be detected. These classifiers are trained with the number of object templates and based on the learning it detects the object instance in the image by matching with each sliding window. Various algorithms exist in the literature which provides efficient results in object detection along with the invariance properties required for good object detection. One of the most successful features describing object for detection is Scale-Invariant Feature Transform (SIFT) [3]. It computes feature points from the images which are invariant to changes in scaling, rotation and lighting conditions. Using the concept of SIFT, different variants improving the invariance limits have been proposed. SIFT is a global approach. Belongie's Shape Context [4] approach is based on extracting object boundary points, and capturing relationship of other shape points to the given point of the shape. This relationship is described using distance and orientation measurements. Binning of the distances and angles is done to generate a histogram. This histogram describes the relationship of each point on the shape to other points. This represents the shape context of the shape. Thus matching two shapes refers finding points having similar shape context on both the shapes. This method has proven efficient for text, finger print recognition, but is sensitive to occlusion. Another approach proposed by Dalal and Triggs is based on using Histograms of Oriented gradients (HOG) [5] as a feature descriptor. It was first used for human detection. In this, shape and local appearance of the object is expressed by local intensity gradients (edges). Image is divided into small regions (blocks) and the histogram of gradients is computed based on the weighted vote of each pixel in the region. Histogram of the each block is the feature descriptor. Illumination invariance is provided by contrast normalization of intensity in each block. This technique does not possess rotation invariance. Local Binary Pattern proposed by Ojala [6] is used for texture classification of gray scale images. Features are generated by assigning labels to each pixel by comparing to its neighboring pixel based on the threshold. Label is 8 digit binary number considering 8 neighbors of the pixel. This technique is illumination and rotation invariant. Apart from the above mentioned state of art techniques, various techniques for object detection have been proposed in different domains. The choice of feature generation for object detection varies with requirement and domain of detection. 1.5 Motivation Object detection has become a significant area of research with the advances in the automation in different fields. Automated Object detection is being applied in varied fields like industry, medical imaging, defense applications, security and surveillance systems, office automations, mobile applications etc. With the emergence of the security systems, automated object detection has become a crucial need. In the recent years, visual surveillance has been an active research topic with its increasing importance of law enforcement, security, and military applications [2]. Automating the task of detection of dangerous and suspicious objects in surveillance images would aid the security staff in their services. Presently the security and surveillance system are human dependent. Images in the security system are monitored by humans to prevent use of dangerous suspicious items. For example, the x-ray scan of the baggage at railway stations, airports etc are monitored by human to detect any unauthorized commodity. Overlooking multiple screens for long time decreases the efficiency of a person as the task of monitoring the images is mundane and repetitive. A study suggest that detection rates for operators monitoring 4, 9 and 16 screens varies around 83%, 84% and 64% respectively, and drops significantly. People almost always outperform software algorithms for detection of object in images. But in the coming years, with the advancement in automation computer would be significantly aid human CCTV (closed-circuit television) operator when it is required to deal with multiple simultaneous videos for continuous hours. Thus it is obvious to automate the detection of suspicious objects for surveillance and security systems. Apart from the above mentioned security system, there is need for visual surveillance in business environments for detecting and recognizing objects of interest [7]. For example, counting and analyzing the crowd in public areas and travel sites, for performing crowd traffic management and product advertisement. Automating the detection process and generating the count for analysis would replace the traditional surveillance systems along with the sparing the man power for other efficient job. Also in the industry, automating the process of locating the electronic components will reduce the manual errors. Identification of products with defects in the manufacturing industry would aid in the production process. In medical imaging, automatic screening of diagnostic images for detection of unwanted tissue formation at the early stage, would lead to the diagnosis and treatment of disease on time. In defense applications, identification of suspicious activities in the critical areas using satellite imageries or real time videos captured using UAV. In Mobile applications, automatic detection of face and facial expressions capturing images. There is lot of research carried out in the field of automatic object detection. Object can be present in different sizes, position and orientation. Detection process finds the instance of object in the images based on the feature descriptor of the template object. It is difficult to store template objects of different size and orientation, which would lead to the huge database and also applying the detection process based on matching with all possible template objects would not be efficient in terms of speed and computation cost. Also it is not appropriate for the real time detection. Thus, it is required to have the automated object detection system which can efficiently detect objects at different scale, position and orientation. Further, existing object detection system use the classifier for detection, these classifiers are trained with number of template objects. However it is difficult to generate the object template database for different objects. This motivates to devise the system which can perform detection using single template object. 1.6 Problem Statement Humans possess ability to quickly identify things just by looking at them. As humans, machines do not possess this intelligence for identifying objects in the scene. For detecting the object, machine need to possess the prior knowledge about the object i.e. model object. Detection process involves extracting the features of the object based on its properties and this extracted feature vector is the basis for detection of object based on feature matching. However, features of the object vary with the variation in illumination, scale, position and orientation. And for efficient detection, machine should be able to identify the object inspite of these variations. Different approaches for generating rotation and scale invariant feature descriptors have been proposed. Among these, SIFT [3] descriptors are invariant to scale, illumination and rotation, but it involves large computation and has large dimension feature vector. HOG [5] technique is invariant to scale and illumination but not rotation invariant. Another technique based on Local Binary Pattern [6], is invariant to scale, rotation and illumination changes, but recognition is slow for large databases. Also the binary data generated is sensitive to noise. Shape Context Descriptor [4] is based on distance and orientation; it finds correspondence of each point of the shape to all other remaining points. The technique is invariant to illumination, scale and rotation but is computationally expensive. Most of the detection system uses learning based classifiers in which the classifiers are trained with different descriptors of object. For detecting the objects at different scale and orientation, one solution would be training the classifier with different size object and at different orientation. However, this would lead to large training dataset. Also creating such database for different categories of object is a cumbersome task. Different algorithms discussed above possess different invariance for object detection. For achieving this invariance, some algorithms generate large size feature vectors and some require large computations. Object shape is the important and effective feature as it is also relevant to human perception. The geometric information possessed by shape remains intact inspite of the change in scale, location, and rotation of the object [8]. The proposed system presents rotation and scale invariant detection of the object based of the shape of the object and the technique is computationally efficient as generated feature descriptor of the is based on the radii of object boundary points at orientation 0 to 360 degree. Also it does not involve creation of database of template object for training the classifier. The detection is performed using single template of the object Hence the problem statement can be stated as: “Proposing Rotation and Scale Invariant Object Detection, which uses single object template for detecting object at different scale and rotation in the scene image based on shape of the object” 1.7 Scope of Work The present system is designed for detection of object instance from the scene image at varying scale, orientation and position. The feature descriptor for the object is extracted based on the shape of the object. The shape of the object is represented using the boundary pixels which are defined by the distance and orientation of the boundary pixels with respect to the centroid of the object. As the shape feature is defined using the object centroid, thus it is not influenced by the surrounding pixels of the image. For providing the translation invariance the object centroid is shifted to origin and all the calculations are performed using centroid as origin. As the object descriptor is based on the radii of the boundary points at regular interval of orientation, the radii varies with the size of the object. In order to provide scale invariance, the radii of each pixel is normalized by dividing with the max radii which then ranges from 0 to 1. Rotation invariance is achieved based on the ordering of the feature vector where maximum radii pixel is taken as the starting point, but for object with non-uniform boundaries the maximum radii point varies. Thus, before creating the feature vector, the target (object in the image) is rotated to align it with the orientation of the object. Then feature vector of normalized radii of size 360 is created for the orientation of 0 to 360 degree. Further detection is performed by matching the feature vector of object and target. The feature vector is based on radii and orientation local to the object. Thus the creation of feature vector is simple and computationally efficient. Also, to overcome the difficulty of creating database of model object for training the classifier the proposed technique is efficient for detection using the single template object. The proposed technique does not consider the illumination changes. The key feature of this technique is its simplicity and detection using single template object. The technique is tested for objects with different shapes at different scale and orientation. 1.8. Organization of Thesis The content of the thesis is organized as follows: Chapter 2: This chapter presents the literature review of existing state of art techniques for object detection. Chapter 3: This chapter describes the proposed system for rotation and scale invariant object detection. Chapter 4: This chapter highlights the implementation details of the system. Chapter 5: This chapter presents the results of the technique for objects with different shapes at different scale and rotation. Chapter 6: The conclusion of the present work along with the future work in described in this chapter CHAPTER 2 LITERATURE REVIEW Object detection is an active research area in the domain of image processing and computer vision. It has become a significant component of modern intelligent systems. Research in this field has led significant advances in field of optical character recognition systems, medical imaging, industrial systems, defence and biometrics. Modern Computer Vision originated in early 1960s. The earliest applications were pattern recognition systems for character. Subsequently, advances in this area proposed solutions in varied fields of object detection making machine independent and executing less human dependent tasks. This chapter analyzes state-of-the-art methods for object detection and description. Object detection usually includes the following parts: data, features and matching. Data are represented by input images. Subsequently, specific type of features is chosen. This type represents different characteristics of objects. In the final stage, the detection process is applied to match the objects based on obtained features and to provide the ability to detect objects in the images. Different Object detection techniques are discussed. 2.1 Scale Invariant Feature Transform Scale Invariant Feature Transform technique is based on local feature descriptors which are invariant to scale, illumination, image noise and little change in perspective. It is proposed by David Lowe in 1999 [3]. 2.1.1 Scale-space extrema detection Scale-space extrema detection is based on detecting key points of all sizes by searching over different scales of the image. Difference-of-Gaussian function is used for identifying interest points which are invariant to scale and orientation. This process involves repeatedly convolving the image with Gaussian functions of $\sigma = \sqrt{2}$ in order to produce a set of scale-space images. This refers to an octave in scale space. Difference of Gaussian images is generated by subtracting adjacent Gaussian images. Size of Gaussian image is halved after each octave, and the process is repeated The scale-space of an image $L(x, y, \sigma)$, which is the convolution of a variable-scale Gaussian, $G(x, y, \sigma)$, with an input image, $I(x, y)$ is given as: (2.1) Where * denotes convolution in x and y direction (2.2) Fig 2.1: Difference-of-Gaussian image pyramid. The difference of two adjacent scales, $D(x, y, \sigma)$, separated by factor k, is given by: (2.3) For finding the set of key points at corresponding scales, each sample point is compared to its eight neighboring pixels and nine pixels in the above and below scale. If the sample point is the maxima or minima amongst the neighboring points, it is considered to be the key point. This process generates set of key points at corresponding scales. This provides scale invariance, as the descriptor of each key point is sampled at the same scale in other image. Fig 2.2: Local extrema detection 2.1.2 Key point Description In Section 2.1.1, a set of key locations and their scales were established. Further for rotational invariance, orientation for each key point is computed. The Gaussian blurred image at the scale corresponding to each key point which provides scale invariance is selected and by taking the pixel difference gradient orientations and magnitudes are calculated. Orientation histogram of 36 bins with orientation of 0 – 360 degree is created. Each sample is added to the histogram based on its gradient magnitude. Histogram peaks signify the dominant directions. (a) (b) Fig 2.3: SIFT descriptor (a) Key point. (b) Descriptor. 2.1.3 Orientation Assignment Invariance to image rotation is provided by assigning orientation to each key point and key point descriptor is represented using this orientation. Using pixel differences, for image $L(x, y)$, gradient magnitude and orientation, $m(x, y)$, and, $\theta(x, y)$, are computed using equation (2.4) and (2.5) (2.4) (2.5) Further, gradient orientations of sample points are added to the orientation histogram. SIFT generates 128 dimensional feature descriptors for each key point. 2.1.4 Key Point

Matching For key point matching, the best match for each key point is found using nearest neighbor approach. The nearest neighbor for each key point is the one with minimum Euclidean distance in the database of keypoints from training images. There exist some features in the image whose correct match is not found in the training database, these features may belong to the background clutter. In order to discard these features, applying threshold on the distance does not give proper results. This technique compares the closest neighbor distance to the second closest neighbor. SIFT technique is based on local information. SIFT possess good stability and invariance, but its high dimensional vector makes the keypoint matching process slow. Different variants of SIFT like Affine SIFT, GSIFT, PCA-SIFT and CSIFT are introduced. Jian et al [9] performed a comparative study of different variants of SIFT which states that SIFT and CSIFT are invariant to scale and rotation change and CSIFT performs better in blur and affine change, but it does not produce good results under illumination change. GSIFT is based on global texture feature and is invariant to illumination change. ASIFT is invariant to affine transformations. PCA-SIFT reduce the dimension of the feature descriptor and give promising results under all situations. SURF is efficient in terms of time complexity, but it is not efficient in terms of detection accuracy. 2.2 Histogram of Gradients Histogram of Oriented Gradients [5] is widely used in pedestrian and vehicle detection. HOG is fast and accurate feature extractor but is not rotation invariant. HOG technique is generally used with the SVM classifier for Object detection. It is proposed by N. Dalal, and B. Triggs in 2005. HOG features are based on local object appearance and shape and are described by the distribution of intensity gradients. Image is divided into small connected regions, called cells, and histogram of gradients for pixels in the each are calculated. Based on the value of gradient, each pixel in the cell casts a weighted vote for orientation histogram. Orientation Histogram varies from 0 to 180 degrees. This cell histogram comprises the feature vector. Image is divided into blocks which consist of cells. Intensity in each block is contrast normalized and this provides illumination invariance. This technique resembles scale-invariant feature transform. The difference between both the techniques is that HOG uses contrast normalization on blocks of the image where blocks correspond to grid of uniformly spaced cells. HOG is based on the idea that the votes of the dominant edge directions of the intensity gradients describe the shapes of the object. This represents the feature descriptor of the object. It is obtained by collecting the histogram of gradient directions for pixels in the cells. As this technique works on localized cells, so invariance to geometric transformations appears in large spatial area. There are different variants of HOG block scheme: Rectangular HOG (R-HOG), Circular HOG (C-HOG), Bar HOG and Center-Surround HOG where R-HOG is the most popular scheme. Fig 2.4: R-HOG block. 2.2.1 Gradient computation (2.6) Computation of image gradients is the first step in HOG feature extraction. Gradients are computed in both horizontal and vertical directions, by applying one dimensional filter kernels as shown below: [-1; 0; 1] and [-1; 0; 1] T Then, for each pixel in I(x, y), the magnitude and orientation is calculated by (2.7) (2.8) Where Gx(x,y) and Gy(x,y) are horizontal and vertical gradient values for each pixel respectively. Figure 2.5: Processing chain of HOG feature descriptor 2.2.2 Orientation binning Histograms for each cell are created and bins of the histogram vary from 0o to 180o (for signed orientation it varies from 0o to 360o), every histogram bin has size of 20o and so there are 9 bins. Each pixel in the cell is assigned to one of the bins based on its orientation. This is weighted voting of the cell. Weight of votes, is based on the gradient magnitude. 2.2.3 Descriptor Blocks For illumination invariance, gradient strength is normalized by grouping the cells into blocks. These blocks are overlapping which facilitates contribution of each cell to the orientation of block more than once. Considering, rectangular block, each block comprises of 2 x 2 cells and each cell consist of 8 x 8 pixels. 2.2.4 Block Normalization Normalization of blocks can be done in different ways which are as follows: (2.9) (2.10) (2.11) Where non-normalized feature vector i.e. histogram for given block is denoted by v, ||vk|| is the k-norm where k = 1, 2 and eps is the small constant. Final HOG feature descriptor consists of the normalized cell histograms for all blocks of the image. Fig. 2.6: Block Normalization Histogram of oriented gradients is invariant to illumination changes and oscillation but this technique is not rotation invariant. On rotation of the object, resulting histogram is also rotated, thus generating different feature vector which leads to failure of object detection. Recently, Fast Circular HOG descriptor is proposed [10] for detecting rotated objects. Conventional Circular HOG consist of circular block and takes as input: number of radial bins and angular bins, radius of center circle and expansion factor for radius. As HOG cannot be directly applied for rotated object detection, features of object after rotation are extracted and this process is to be conducted for each window which is not efficient in real time detection. Fast circular HOG technique uses a shift matrix which when multiplied to HOG feature vector generates the rotated feature vector for each window and is faster as compared to conventional HOG for rotated object detection. This technique is not efficient in terms of accuracy but it reduces the time of execution. 2.3 Shape Context Descriptor Belongie, Malik and Puzicha proposed the technique for object recognition based on shapes matching using a nearest neighbor approach in 2002 [4]. This technique, introduced a new descriptor for representing the shape of the objects which is referred to as Shape Context. It is a global descriptor based on contour. In this descriptor, relative position of each object contour point to all other contour points of the shape is represented in the form of histogram. For each point ai on the object shape, corresponding best match bj is found on the target shape using Euclidean distance and the angle from each point to all other n-1 points. Binning is performed after computing log of the distance, distance and angle measurements. Consider, object contour consists of n points, for point ai, histogram of relative position of remaining (n-1) points is created which is given as: (2.12) gi (k) = { b ≠ ai : (b - ai) Є bin (k) } The histogram gi (k) in the above equation is the shape context of point ai . This technique uses log polar coordinates. The cost of matching points is given by: (2.13) Where g(k) and h(k) denotes the k bin histogram at point a in the object and b in the target shape respectively. Following figure shows the process of generating Shape Context descriptor. (a) (b) (c) (d) (e) (f) Fig 2.7: Shape Context Descriptor: a) and b) points on the boundary of object. (c) Histogram creation using log-polar bins. d) shape context for point in a). e) Shape context for point in b)and f) shape context for the left point in b) These descriptors form an efficient method of matching the shapes of objects in images. Once the cost of matching the points is found, this cost is minimized using Hungarian Method. This problem of minimization of cost is considered as the square assignment or bipartite graph matching problem. This technique is invariant to scaling and translation. Rotation invariance is provided by considering the tangent vector at each point as the positive x axis which rotates the reference frame with the tangent angle and the resulting descriptor is rotation invariant. This process of rotation invariance has high complexity. Several variations in the shape context descriptor are introduced [8]. Inner Distance shape Context overcomes the problem of handling articulated shapes which exist in generalized shape context. Inner distance i.e. shortest distance between shape points is considered in place of Euclidian distance. Another variation is the Shape Context using Indexing, which increases the efficiency of shape context in terms of speed by reducing the matching candidates using standard deviation and mean distance of shape context for matching. High complexity of Rotation invariance in generalized shape context technique is reduced in FFT RISC i.e. Rotation Invariant Shape Context using FFT. 2.4 Local Binary Pattern Descriptor In 2002, T. Ojala, M. Pietikainen, and T. Maenpaa, proposed the concept of Local Binary Patterns for texture classification of gray scale images which is invariant to rotation [6]. These operators are robust to changes in gray scale which provides illumination invariance. This texture operator assigns the label to each image pixel by thresholding its neighborhood. This label is the binary number. The window is divided into cells and each pixel in the cell is compared to its neighboring pixels along a circle in clockwise or anti- clockwise direction as shown in Fig 2.8. Fig. 2.8: Neighborhood for calculation of LBP 8 digit binary number is generated by comparing the centre pixel to the 8 neighboring pixels, if neighbor pixel is less than the centre pixel, digit is assigned zero and if the neighbor pixel is greater than the centre pixel, digit is assigned one. This binary number is converted to decimal and histogram is computed over the cell based on the frequency of occurrence of the number. This generates the feature vector of size 256. Histogram of each cell is concatenated to form the feature vector of the window. As the size of the image patches whose histograms are compared varies, the histograms are normalized for coherent description. Following figure shows the creation of LBP descriptor for face [11]. Fig. 2.9: Creation of LBP descriptor for Face The gray scale invariance is achieved by subtraction the centre pixel from each neighboring pixel and the rotation invariance is achieved by performing circular bitwise right shift. This technique is invariant to illumination and rotation. Once the feature descriptor for all rotation is generated the classifier is trained and the detection can be carried out. LBP is invariant to illumination, rotation and scale but it has some shortcomings [12]. Due to long histograms, recognition speed is slow on large-scale database. Also due to non consideration of the effect of centre pixel, local structure can be missed. Produced binary data is sensitive to noise. 2.5 Dominant Edge Directions In 2015, Marcin Kmiec and Andrzej Glowacz proposed the technique for detecting the specific class of objects, i.e. knives for security applications. The basis for this technique is the shape of the knife which constitutes of dominant edges. It is not the state of art technique as it is confined to specific shape objects only. Here the shape is described based on the orientation of the edges with respect to the vertical axis of the image. Since the knife blade has a longitudinal shape, detecting the dominant edge direction consistent with the detection window's vertical axis could constitute a possible knife candidate. First, edges are extracted from the image and divided into fixed length segments. These segments are approximated with a straight line segment and can be of any shape. After approximation of edges, orientation of each segment with respect to the vertical axis of detection window is calculated. This orientation is used to filter the edges other than the edges contributing the object shape. To filter those edges a threshold value for orientation is defined. Edges having orientation below threshold define the object shape. This technique takes the fixed image size is 64 × 128 pixels for defining the size of feature vector. If the image is larger in size, it is scaled down to the base size. For creation of feature vector for object shape description, an auxiliary image of size 64 x 128 is created and pixel intensity of filtered segments defining object shape is calculated based on the orientation of the segment. The pixel intensity of segments is calculated using following formula: (2.14) Where S denotes the angular orientation of the segment and Sx, y is the angular threshold. (a) (b) (c) (d) (e) Fig. 2.10: Generation of Object Feature using Dominant Edge Direction, (a) Object, (b) Edge map, (c) Approximation of edges to segments (d) Filtered segments after applying threshold (e) Object segments with calculated intensity. The final feature vector F is created using the intensity of pixels in the auxiliary image created after filtering the segments and intensity calculation and is generated by adding its consecutive rows. After creating the feature vector, Linear SVM classifier is used for detection of object. This technique is not rotation invariant as the shape is described based on orientation of object with respect to vertical image axis. For providing rotation invariance, the detection window is rotated at regular orientation and feature vector is generated at every orientation for detection of object. This increases the time for detection. To overcome this, the proposed system uses the rotation invariant technique based on orientation of the boundary pixels of the object. CHAPTER - 3 PROPOSED SYSTEM This chapter presents the proposed system for object detection which is invariant to transformation, rotation and scaling. Detection of object at different orientation and scale in the image using a single template model is a challenging task. Objects are distinguishable on the basis of their visible features, and among these features, shape of the object is an important key for its recognition. The proposed detection system is based on the shape of the object. 3.1 Brief Description: Rotation and Scale Invariant Object Detection Object detection process in the proposed system consists of three phases: Segmentation and Boundary Extraction phase, Shape Representation and Orientation Alignment phase and finally Object Detection using Shape Matching. The first and foremost step for object detection is segmentation which is carried out using Active Contour [13]. After segmentation the boundary extraction on the segmented is carried out. Thereafter, these extracted boundary points which represent the shape of the object are used for the creation of scale and rotation invariant feature vector. Shape representation in done using polar transformation (distance and orientation) of the boundary points about the geometric centre of the object [14]. For scale invariance the normalization of the feature vector is done. And for rotation invariance, the target in the scene image is aligned with the orientation of the object. The alignment of orientation is carried out using the local extrema vertices of the object and the target. After normalization and orientation alignment, the feature vector of the object and target are generated. Further detection of object is carried out by shape matching using the feature vector and also for concave objects computing cost using nearest neighbor approach for boundary pixels. The proposed system uses the single template object. Sliding window is used for the detection of object in the scene image. Feature vector for each window is generated and matching is performed. 3.2 Flow of the System Fig. 3.1 Flow of the System Figure 3.3 represents the flow of the system. The detailed explanation of the steps is given in subsequent sections. 3.3 Algorithm The algorithm for the proposed system is described below: Step1: Segment the input image (object and target) and extract boundary pixels. Step2: Compute centroid and transform the boundary pixels by shifting the centroid to origin Step3: Generate the distance (radii) and orientation of each boundary pixel with respect to centroid. Step4: Normalize the distance vector with the max radius and order the distance and orientation vector taking max radii point as the starting point. Step5: Calculate First Order Derivative and find all maxima and minima points of the boundary. Step6: Compute equal number of distinctive vertices for object and target using the first order derivative of extrema points of boundary. Step7: Using the polar representation of distinctive vertices of the object and target, compute the optimal rotation angle. Step8: Rotate the target with the optimal rotation angle. Step9: Calculate the feature vector for both object and target by calculating the radii for the boundary pixels having orientation from 0° to 360° Step10: Perform Shape matching using Feature Vector. 3.4 Segmentation and Boundary Extraction Segmentation is the preliminary step of almost all object detection techniques. In computer vision, image segmentation refers to separating multiple segments present in the image for simple representation and easy analysis. Segmentation process facilitates locating the boundaries of the objects in images. In this system, the segmentation is carried out using Active Contour Model without Edges proposed by Chan and Vese [13]. Fig. 3.2 Segmentation and Boundary Extraction The above mentioned Active Contour model for image segmentation is based on techniques of curve evolution, Mumford-Shah functional model for segmentation and level sets [16]. The key feature of this model is that the detection of boundaries is not dependent on gradient. The model is energy minimization based segmentation. The model considers the image u0 formed by two regions of piecewise-constant intensities, with distinct values u0i and u0o. The object to be detected is represented by the region with the value u0i and its boundary is represented by C. The fitting energy is defined as: (3.1) where C is boundary of an open and bounded domain , c1 = average(u0) inside C , c2 = average(u0) outside C and |C| is the length of the boundary curve C Fig. 3.3: Different position of the curve and fitting energy The above figure describes that the fitting energy is minimized only for the case when the curve is on the boundary of the object. In proposed system, the initial curve in the form of mask is defined based on the dimensions of the image and segmentation is carried using this initial mask. Segmentation process returns the multiple segments of the image which represent the foreground with white pixels and background as black. For boundary tracing starting pixel should be the part of foreground, image is scanned till the starting pixel is found. Thereafter extraction of boundaries of these segments is carried out which returns the boundary pixels of the segmented objects in the image. 3.5 Shape Representation and Orientation Alignment In order to provide representational invariance to position, scale and rotation, the geometric centre of extracted shape boundary is selected as a reference point. The coordinates of the centroid is calculated using the following equation: (3.2) Where, n is the number of boundary pixels The proposed system represents shape of object based on distance and orientation of boundary pixels. Features of object shape from centroid of object are calculated as: (3.3) (3.4) Where, ∆x and ∆y is the difference of coordinates of boundary pixels with respect to centroid of the object. The shape of the object is represented using the feature vector of distance at equal interval of orientations of boundary pixels with respect to centroid. The size of feature vector depends on the interval of orientation in degrees where orientation varies from 0 to 2π. The process for feature vector generation is shown in fig 3.4. Fig 3.4: Rotation and Scale Invariant Feature Generation For transformation invariance, all the boundary pixels are centered i.e. the centroid is shifted to origin and accordingly, all boundary pixels coordinates are changed. For scale invariance, the distance corresponding to each boundary pixel is divided by the maximum distance which ranges from 0 to 1. The feature vector created after normalizing the distance is scale invariant. For rotation invariance, the starting point of the feature vector is the maximum distance boundary pixel and starting from this point, the orientation of boundary pixels is calculated at the fixed interval and the corresponding radii is added to the feature vector. As the starting point of the feature vector creation is based on max radii pixel and the subsequently, pixels contributing to feature vector are based on the orientation of the maximum radii pixel at fixed orientation interval with respect to the orientation of the max radii. This ordering of the feature vector contributes to rotation invariance. Fig. 3.5 Shape Representation: using distance and angle of the boundary points with respect to centre Creating rotation invariant feature vector using single max radii point is considered to be the ideal case. As it is not always true that the maximum radius will always occur at the same point and also for the symmetric shapes there can be more than one maximum radii point. In order to deal with such likely event of objects which have different maximum radii, it is required to consider all the candidate maximum radii points 3.5.1 Rotation Invariance with Multiple Maximum Radii Points In order to perform rotation invariant detection, it is required to align the orientation of the target in the test image with the orientation of the object. This process of aligning the object and target at same orientation requires same number of vertices. These vertices are the set of scale independent, object specific reference points i.e. the local minima and maxima points of the object shape. These are extracted by a smoothed first order derivative on the polar representation of the object boundary points and by finding the zero crossing of the resulting derivative. The smoothed first order derivative of the object boundary is calculated using following formula: (3.5) Where θp+i is ith value of orientation of θ after θp, θp-i is the ith orientation value preceding θp. Rp is the normalized radius i.e. fraction of maximum radius at angle θp and n is the smoothing factor. The scope of the smoothing denoted by n specifies the localness of the minima and maxima which corresponds to the vertices of the object to be used for process of alignment. Finding the vertices reduces the varying number of boundary points between object and target from very high to a very low and distinctive number of vertices. The number of vertices representing the model object determines the required number for aligning orientation and the smoothing factor is iteratively adjusted till an equal number of vertices are found for the target object which is unknown. Further, once the similar number of vertices is found, the angle of rotation with which the target should be rotated in order to align it with the orientation of object is calculated. Rotating the target with the angle of rotation, then leads to minimization of Euclidean distance between the corresponding vertices. The optimal rotational angle is calculated using: (3.6) Where Φ denotes the angle of rotation for alignment, θp+s denotes orientation of vertex (p + s) modulo P, normalized radius of vertex p is denoted by Rp , P is number of vertices, and s is the offset for optimal correspondence of vertices. Finally, rotating the target with optimal rotation angle aligns the orientation of object with the target. After alignment the matching of the shapes is performed. 3.6 Object Detection using Shape Matching Once the orientation of the object and target are aligned, matching of the object and target shapes can be carried out. The shape is described using the distance of the boundary points and orientation with respect to the centroid of the object. Feature vector of the radii at specific orientation interval is created and further object and target are matched using these feature vectors. Size of the feature vector depends on the interval of orientation. The orientation interval is taken as 1° and the size of feature vector is 360. Matching cost is calculated using the sum of squared differences between the feature vectors. Shape matching using feature vector is efficient in case of convex shape objects, but this technique does not perform efficient detection of object in case of concave objects as there can be multiple boundary pixels with same orientation and different radii which can be missed during feature vector creation. This can lead to shape mismatch. For matching shapes with concave boundary, cost of nearest neighbor for each boundary pixel in the object to that in the target boundary is found. Object and target with lower cost are of similar shapes. Detection of object in the scene image is carried out using the sliding window. For each window, the feature vector is created using the above mentioned procedure and the matching for the object feature vector and feature vector for the window is carried out. CHAPTER – 4 IMPLEMENTATION This chapter describes the implementation details of the Rotation and Scale Invariant Object detection process. The process consists of three phases. Initial phase is the segmentation phase and extraction of boundary of segmented object. Second phase is shape representation phase in which model object and target to be detected are aligned at the same orientation, this phase generates the representation of the shape which is invariant to transformation, scale and rotation. Final phase is the detection phase using the sliding window, where each sliding window is matched with the model object for detection of target. The system is developed using Matlab R2013a platform. 4.1 Segmentation and Boundary Extraction Implementation Details As mentioned in the previous chapter the segmentation process is carried out using the Active Contour without Edges proposed by Chan and Vese [13]. It segments the 2-D grayscale image into foreground (object) and background regions. The output image is the binary image in which the foreground is white (logical true) and the background is black (logical false). It takes as input, a mask which is a binary image that specifies the initial state of the active contour. The initial contour position used for contour evolution to segment the image, is defined by boundaries of the mask. Here, the mask is taken closer to the boundary of the image. It also takes the maximum number of iterations to be performed for segmentation. After segmentation, boundaries of the segmented objects are extracted and these boundary pixel coordinates are used for further processing in shape representation. Fig. 4.1 Segmentation using Active Contour 4.2 Implementation: Orientation Alignment and Shape Representation Object shape is represented by discrete set of points which are the external contour points of the object. These set of points vary for object and target to be searched. The proposed approach represents the shape, based on the radii and orientation of the boundary points. As mentioned is the previous chapter, the centroid of the model object and the target is shifted to origin and accordingly the boundary pixels are also translated to provide translation invariance. The distance for each boundary pixel from the centroid is normalized by dividing each distance with the maximum distance. The distance lies between 0 and 1 and this normalization provides scale invariance. For rotation invariance, the starting point of the feature vector describing the shape of object is the maximum distance point. Starting from the max distance point, distance of the boundary pixel having orientation at specific interval (e.g. 1 degree or 10 degree) is stored in the feature vector. As the starting point is always the max distance point, and distance at regular interval orientation constitutes the feature vector of the object shape, this provides rotation invariance to the object. The keypoint for the rotation invariance is the starting point (max distance). But, this is true for ideal convex shapes. For other complex, concave shape object, certain boundary pixels which are significant for object shape description are missed. For example, considering a square or a pentagon which are ideal convex shapes, generating and matching the feature vector based on orientation and distance taking max distance point as starting point is the easier task as shown in Fig 4.2. Fig. 4.2 Shape Representation and Feature Vector Creation for Standard Convex Shapes In the above figures, where the standard convex objects square and pentagon are taken, feature vectors for the object and the rotated objects are similar radii point always which facilitates shape matching using feature vector. But in case of complex concave objects, the feature vector varies as the max radii points which are taken as starting point for feature vector creation are different and thus there is shape mismatch. Fig. 4.3 Shape Representation and Feature Vector Creation for Complex Concave Shapes 4.2.1 Implementation: Alignment of Orientation of Object and Target In order to overcome this problem of shape mismatch for similar shape objects with different orientation, it is required to align the orientation of the object and the target. For this process of alignment, the set of scale independent, object specific reference points i.e. the local minima and maxima points of the object and target shape are extracted. These are extracted by a smoothed first order derivative on the polar representation of the boundary points and then by finding the zero crossing of the resulting derivative as shown in the Fig 4.4. (a) (b) (c) (d) (e) (f) (g) (h) Fig. 4.4 Normalized Radii and First Order Derivative of Object and Target (a) Object, (b) Target , (c) and (d) Normalized Radii of Object and Target respectively , (e) and (f) First Order Derivative of with zero crossing points of Object and Target respectively , (g) and (h) Normalized Radii and First Order Derivative of Object and Target respectively on same plot Calculating the zero crossing points of the First Order Derivative gives the set of vertices which are required to calculate the Optimal Rotation Angle for orientation alignment. (a) (b) Fig. 4.5 (a) and (b) Different Distinctive vertices of Object and Target i.e. set of vertices for calculation of Optimal Rotation Angle After finding the equal number of vertices for Object and Target, Optimal Rotation angle with which the Target should be rotated in order to align it with the orientation of the model Object is calculated. The target is rotated with Optimal Rotation angle as shown in Fig 4.6 Fig. 4.6 Object, Target before and after rotation 4.3 Detection of Object using Shape Matching Detection process uses the sliding window over the image where target is to be detected. For each sliding window, segmentation and boundary pixels are extracted. And further set of vertices for orientation alignment are computed using first order derivative and the smoothing factor n is varied in order to find the equal number of vertices. After aligning the target with the orientation of the object, cost of matching the shapes of the object and target is computed. For this the feature vector of radii for 0 to 360 degree orientation is computed. Object and Target with similar shape has similar feature vector. (a) (b) (c) Fig. 4.7 Feature vector (a) and (b) Computation of radii at 0 to 360 degree orientation for Object and Target boundary (c) Final Feature vector for Object (red) and Target (green) For computing the cost of matching the feature vectors, Euclidean distance between the feature vectors is calculated and the minimum cost feature vectors are the best shape match. Finally the window having the minimum cost is the result of detection of the target. But for certain objects, while computing the feature vector some boundary pixels are missed due to overlapping orientation. In this case the cost of matching the object and the target is computed using the nearest neighbor. In this, for each sliding window, after orientation alignment, Euclidian distance between the object boundary pixel and the corresponding nearest neighbor in the target boundary is calculated. Object and Target with similar shapes have minimum nearest neighbor cost.

CHAPTER - 5 RESULTS This chapter presents the results of the proposed model for rotation and scale invariant object detection. The system is evaluated for different set of objects. The proposed technique is computationally efficient. Results of detection are based on the sliding window approach. 5.1 Detection of Complex Shape Object: Aircraft Result of detection for multiple instances of Aircraft in an aerial photograph along with the template object is shown in figure 5.1and 5.2. The different aircrafts positioned in the image are placed at different orientation and are detected successfully. Fig 5.1: Template Object: Aircraft Fig 5.2: Detection Results: Aircraft The size of the template object is 90 x 110 and that of target in the scene image is 40 x 40. The cost of matching is 0.08. 5.2 Detection of Natural Object: Leaf Following figure shows the template object and the result of detection for natural object Leaf in the cluttered image. Fig 5.3: Template Object: Leaf Fig 5.4: Detection Results: Leaf The size of the template object is 200 x 188 and that of the target in the scene image is 110 x 90. The cost of matching is 0.05. CHAPTER 6 CONCLUSION AND FUTURE WORK This chapter states the conclusion derived from the present research work and discusses the future work for the proposed system. 6.1 Conclusion In today's digital world, image processing in general and object detection in particular has gained significant importance. It has become the buzzword in varied sectors, like industry sector, medical sector, military scenario and it is also being used for security settings. Automating the process of detection from image facilitates efficient use of manpower for different purposes rather than performing the mundane task of monitoring images and reduces the manual errors. Detection of object in an image is based on matching the object instances with the template object. Object appears at different size and orientation, detecting the object with variable sizes and orientation is the challenging task. The proposed approach attempted to detect the multiple instances of the object with varying scale and orientation. The proposed algorithm is computationally efficient and simple to implement. Object is represented based on its shape. Feature descriptor of the object is created using the normalized distance of the object boundary at orientation varying from 0 ° to 360 °. The orientation of the target in the image is aligned with the template object and feature vector is generated and finally the matching is performed using feature vector. For the complex concave objects, nearest neighbor distance for the corresponding boundary pixels of object and target is used. Detection of object in the image is performed using sliding window. The proposed system is successfully tested on objects of different shapes placed at varying size and orientation. This technique requires less computation and successfully detects the object in small images, for large size images, the time required for detection need to be improved. 6.2 Future Work The present technique produces promising results for detection of the objects with varying size and orientation. It is based on the shape of the object; the complete shape of the object is taken as the basis for detection. Partially occluded objects in the image are not detected. Also, the present technique does not consider the objects captured at different viewing angles. Hence the technique can be improved for detection of partially occluded objects and the objects at different view angles.

[Metadata removed]

Hit and source - focused comparison, Side by Side:

Left side: As student entered the text in the submitted document.
Right side: As the text appears in the source.