ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

A Survey on Object Detection and Tracking Algorithms

Subhash Nevhal[1], Dr.Archana Ghotkar[2]

M.E. Student, Dept. of Computer Engineering, Pune Institute of Computer Technology, Pune, India[1].

Associate Professor, Dept. of Computer Engineering, Pune Institute of Computer Technology, Pune, India[2].

ABSTRACT: Object tracking segments is a region of interest from a video scene and keeps track of its motion, positioning and occlusion. The object detection and object classification are required for tracking an object in sequence of images. Object detection and tracking are important and challenging tasks in many computer vision applications such as surveillance, vehicle navigation, and autonomous robot navigation. Every tracking method requires an object detection mechanism either in every frame or when the object first appears in the video. Our survey is focused on the various methods that have been used for detection and tracking of the objects.

KEYWORDS: Image processing and Computer Vision; Object detection; Tracking; point tracking; Kernel Tracking; Silhouette Tracking; contour evolution; feature selection; shape tracking

I. INTRODUCTION

Object tracking is an important task within the field of computer vision. Tracking can be defined as the problem of estimating the trajectory of an object in the image plane as it moves around a scene. A tracker assigns consistent labels to the tracked objects in different frames of a video. Depending on the tracking domain, a tracker can also provide object-centric information, such as orientation, area, or shape of an object. The suitability of a particular tracking algorithm depends on object appearances, object shapes, number of objects, object and camera motions, and illumination conditions. In a tracking scenario, an object can be defined as anything that is of interest for further analysis. Objects can be represented by their shapes and appearances. We will first describe the object shape representations commonly used for tracking.

- Points: The object is represented by a point, that is, the centroid or by a set of points. The point representation is suitable for tracking objects that occupy small regions in an image.
- Primitive geometric shapes: Object shape is represented by a rectangle, ellipse etc.
- Object silhouette and contour: Contour representation defines the boundary of an object. The region inside the contour is called the silhouette of the object. Silhouette and contour representations are suitable for tracking complex non-rigid shapes
- Articulated shape models: Articulated objects are composed of body parts that are held together with joints.
- Skeletal models: Object skeleton can be extracted by applying medial axis transform to the object silhouette. Skeleton representation can be used to model both articulated and rigid objects.

There are a number of ways to represent the appearance features of objects.

- Probability densities of object appearance: This can either be parametric, such as Gaussian and a mixture of Gaussians or nonparametric, such as Parzen windows and histograms. The probability densities of object appearance features (color, texture) can be computed from the image regions specified by the shape models.
- Templates: Templates are formed using simple geometric shapes or silhouettes. An advantage of a template is that it carries both spatial and appearance information. Templates, however, only encode the object appearance generated from a single view. Thus, they are only suitable for tracking objects whose poses do not vary considerably during the course of tracking.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

- Active appearance models: Active appearance models are generated by simultaneously modeling the object shape and appearance. The object shape is defined by a set of landmarks. Similar to the contour-based representation, the landmarks can reside on the object boundary; otherwise they can reside inside the object region. For each landmark, an appearance vector is stored which is in the form of color, texture, or gradient magnitude. Active appearance models require a training phase where both the shape and its associated appearance learned from a set of samples using, for instance, the principal component analysis.
- Multi-view appearance models: These models encode different views of an object. One approach to represent the different object views is to generate a subspace from the given views. Subspace approaches, for example, Principal Component Analysis (PCA) and Independent Component Analysis (ICA), have been used for both shape and appearance representation.

There is a strong relationship between the object representations and the tracking algorithms. Object representations are usually chosen according to the application domain. For tracking objects, which appear very small in an image, point representation is used.

II. FEATURE SELECTION FOR TRACKING

Feature selection is closely related to the object representation. For example, color is used as a feature for histogram-based appearance representations, while for contour-based representation, object edges are usually used as features. Many tracking algorithms use a combination of these features. The details of common visual features are as follows.

- Color: The apparent color of an object is influenced mainly by two physical factors, 1) the spectral power distribution of the illuminant and 2) the surface reflectance properties of the object. In image processing, the RGB (red, green, blue) color space is usually used to represent color. However, the RGB space is not a perceptually uniform color space, that is, the differences between the colors in the RGB space do not correspond to the color differences perceived by humans. $L^* u^* v^*$ and $L^* a^* b^*$ are perceptually uniform color spaces, while HSV (Hue, Saturation, Value) is an approximately uniform color space. These color spaces are sensitive to noise. Bashir Muhammad[1] proposed a skin detection method combining two color spaces HSV (Hue, Saturation, Value) and YCgCr (luminance, chrominance in green, chrominance in red). The S, Cg and Cr components are used to form a hybrid SCgCr color space. Skin detection results show that, the method can respond well to different skin color tones with less sensitivity to skin-like background pixels.
- Edges: Object boundaries generate strong changes in image intensities. Edge detection is used to identify these changes. An important property of edges is that they are less sensitive to illumination changes compared to color features. Algorithms that track the boundary of the objects usually use edges as the representative feature. Because of its simplicity and accuracy, the most popular edge detection approach is the Canny Edge detector. Juseong Lee et al. [2] present an energy-efficient architecture of the cannyedge detector for advanced mobile vision applications. Three key techniques for reducing computational complexity of the Canny edge detector are presented.
- Optical Flow: Optical flow is a dense field of displacement vectors which defines the translation of each pixel in a region. It is computed using the brightness constraint, which assumes brightness constancy of corresponding pixels in consecutive frames. Optical flow is commonly used as a feature in motion-based segmentation and tracking applications.
- Texture: Texture is a measure of the intensity variation of a surface which quantifies properties such as smoothness and regularity. Compared to color, texture requires a processing step to generate the descriptors. There are various texture descriptors: Gray-Level Co-occurrence Matrices (GLCM's). Similar to edge features, the texture features are less sensitive to illumination changes compared to color. [3] The first step carries out a segmentation of the image pixels based on their "color". They associated a suitable feature vector with each fragment and vectors related to fragments of the same texture form a well-defined cluster in the feature space and covered complete textures by merging together groups of fragments.

Mostly features are chosen manually by the user depending on the application domain. However, the problem of automatic feature selection has received significant attention in the pattern recognition community. Automatic feature selection methods can be divided into filter methods and wrapper methods. The filter methods try to select the features

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

based on a general criteria, for example, the features should be uncorrelated. The wrapper methods select the featuresbased on the usefulness of the features in a specific problem domain, for example, the classification performance using a subset of features. Principal Component Analysis (PCA) is an example of the filter methods for the feature reduction.

III. OBJECT DETECTION

A common approach for object detection is to use information in a single frame. However, some object detection methods make use of the temporal information computed from a sequence of frames to reduce the number of false detections. This temporal information is usually in the form of frame differencing, which highlights changing regions in consecutive frames. Given the object regions in the image, it is then the tracker's task to perform object correspondence from one frame to the next to generate the tracks.

- Point Detectors:

  Point detectors are used to find interest points in images which have a texture in their respective localities. Interest points have been long used in the context of motion, stereo, and tracking problems. A desirable quality of an interest point is its invariance to changes in illumination and camera viewpoint. A desirable quality of an interest point is its invariance to changes in illumination and camera viewpoint. In the literature, commonly used interest point detectors include Moravec's interest operator [4], Harris interest point detector [5], KLT detector [6], and SIFT detector [7]. For a comparative evaluation of interest point detectors, we refer the reader to the survey by Mikolajczyk and Schmid [8]. Moravec's operator computes the variation of the image intensities in a 4×4 patch in the horizontal, vertical, diagonal, and anti-diagonal directions and selects the minimum of the four variations as representative values for the window. A point is declared interesting if the intensity variation is a local maximum in a 12×12 patch. The Harris detector computes the first order image derivatives, (I_x , I_y), in x and y directions to highlight the directional intensity variations, then a second moment matrix, which encodes this variation, is evaluated for each pixel in a small neighborhood. Both Harris and KLT emphasize the intensity variations using very similar measures. For instance, R in Harris is related to the characteristic polynomial used for finding the eigenvalues of M: $\lambda^2 + \det(M) - \lambda \cdot \mathrm{tr}(M) = 0$, while KLT computes the eigenvalues directly. In order to introduce robust detection of interest points under different transformations, Lowe [7] introduced the SIFT (Scale Invariant Feature Transform) method which is composed of four steps. First, a scale space is constructed by convolving the image with Gaussian filters at different scales. Convolved images are used to generate difference-of-Gaussians (DoG) images. Candidate interest points are then selected from the minima and maxima of the DoG images across scales.
- Background Subtraction:

  Object detection can be achieved by building a representation of the scene called the background model and then finding deviations from the model for each incoming frame. Any significant change in an image region from the background model signifies a moving object. The pixels constituting the regions undergoing change are marked for further processing. Usually, a connected component algorithm is applied to obtain connected regions corresponding to the objects. This process is referred to as the background subtraction. Background subtraction became popular following the work of Wren et al. [9]. Stauffer and Grimson [10] use a mixture of Gaussians to model the pixel color. In this method, a pixel in the current frame is checked against the background model by comparing it with every Gaussian in the model until a matching Gaussian is found. If a match is found, the mean and variance of the matched Gaussian is updated, otherwise a new Gaussian with the mean equal to the current pixel color and some initial variance is introduced into the mixture. Each pixel is classified based on whether the matched distribution represents the background process.
- Segmentation:

  The aim of image segmentation algorithms is to partition the image into similar regions. Every segmentation algorithm addresses two problems, the criteria for a good partition and the method for achieving efficient partitioning. Mean-Shift Clustering: For the image segmentation problem, Comaniciu

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

[11] propose the mean-shift approach to find clusters in the joint spatial+colour space, [l , u, v, x, y], where [l , u, v] represents the colour and [x, y] represents the spatial location. Given an image, the algorithm is initialized with a large number of hypothesized cluster centers randomly chosen from the data. Then, each cluster center is moved to the mean of the data lying inside the multidimensional ellipsoid centerd on the cluster center. The vector defined by the old and the new cluster centers is called the mean-shift vector. The mean-shift vector is computed iteratively until the cluster centers do not change their positions.

- Image Segmentation Using Graph-Cuts:
  Image segmentation can also be formulated as a graph partitioning problem, where the vertices (pixels), V = {u, v, . . .}, of a graph (image), G, are partitioned into N disjoint sub-graphs (regions), A_i , $U^N_{i=1}$ $A_i = V$, $A_i \cap A_j$ = □, $i \neq j$, by trimming the weighted edges of the graph. The total weight of the trimmed edges between two sub-graphs is called a cut. The weight is typically computed by color, brightness, or texture similarity between the nodes. Shi and Malik [12] propose the normalized cut to overcome the over segmentation problem. In their approach, the cut not only depends on the sum of edge weights in the cut, but also on the ratio of the total connection weights of nodes in each partition to all nodes of the graph.
- Active Contours:
  In an active contour framework, object segmentation is achieved by evolving a closed contour to the object's boundary, such that the contour tightly encloses the object region. Evolution of the contour is governed by an energy functional which defines the fitness of the contour to the object region.
- Supervised Learning:
  Object detection can be performed by learning different object views automatically from a set of examples by means of a supervised learning mechanism. Learning of different object views do not required of storing a complete set of templates. Given a set of learning examples, supervised learning methods generate a function that maps inputs to desired outputs. A standard formulation of supervised learning is the classification problem where the learner approximates the behavior of a function by generating an output in the form of either a continuous value, regression, or a class label, which is called classification. In context of object detection, the learning examples are composed of pairs of object features and an associated object class where both of these quantities are manually defined. Selection of features plays an important role in the performance of the classification; hence, it is important to use a set of features that differentiate one class from the other. Once the features are selected, different appearances of an object can be learned by choosing a supervised learning approach. These learning approaches include neural networks, adaptive boosting [13], decision trees, and support vector machines etc. Supervised learning methods usually require a large collection of samples from each object class. This collection must be manually labeled. A possible approach to reducing the amount of manually labeled data is to accompany co-training with supervised learning.
- Adaptive Boosting:
  Boosting is an iterative method of finding a very accurate classifier by combining many base classifiers, each of which may only be moderately accurate. In the training phase of the Adaboost algorithm, the first step is to construct an initial distribution of weights over the training set. The boosting mechanism then selects a base classifier that gives the least error, where the error is proportional to the weights of the misclassified data. Next, the weights associated with the data misclassified by the selected base classifier are increased. Thus the algorithm encourages the selection of another classifier that performs better on the misclassified data in the next iteration [14].
- Support Vector Machines:
  As a classifier, Support Vector Machines (SVM) are used to cluster data into two classes by finding the maximum marginal hyper-plane that separates one class from the other. The margin of the hyper-plane, which

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

is maximized, is defined by the distance between the hyper-plane and the closest data points. The data points that lie on the boundary of the margin of the hyperplane are called the support vectors. SVM can also be used as a nonlinear classifier by applying the kernel trick to the input feature vector extracted from the input. The kernels used for kernel trick are polynomial kernels or radial basis functions [14].

IV. OBJECT TRACKING

The aim of an object tracker is to generate the trajectory of an object over time by locating its position in every frame of the video. Object tracker may also provide the complete region in the image that is occupied by the object at every time instant. The tasks of detecting the object and establishing relation between the object instances across frames can either be performed separately or jointly. In the first case, possible object regions in every frame are obtained by means of an object detection algorithm, and then the tracker correspond objects across frames. In the latter case, the object region and correspondence is jointly estimated by iteratively updating object location and region information obtained from previous frames. The main tracking categories are:

- Point Tracking:
  Tracking can be formulated as the correspondence of detected objects represented by points across frames. Point correspondence is a complicated problem especially in the presence of occlusions, mis-detections, entries, and exits of objects. In General, point correspondence methods can be divided into two broad categories, namely, deterministic and statistical methods. The deterministic methods use qualitative motion heuristics [15] to constrain the correspondence problem. On the other hand, probabilistic methods explicitly take the object measurement and take uncertainties into account to establish correspondence.

1. Deterministic Methods for Correspondence: Deterministic methods for point correspondence define a cost of associating each object in frame $t - 1$ to a single object in frame t using a set of motion constraints. Minimization of the correspondence cost is formulated as a combinatorial optimization problem. A solution, which consists of one to-one correspondences. The correspondence cost is usually defined by using a combination of the following constraints.
   - Proximity assumes the location of the object would not change notably from one frame to other.
   - Maximum velocity defines an upper bound on the object velocity and limits the possible correspondences to the circular neighborhood around the object.
2. Statistical Methods for Correspondence: Measurements obtained from video sensors invariably contain noise. Statistical correspondence methods solve these tracking problems by taking the measurement and the model uncertainties into account during object state estimation.
   - Single Object State Estimation:
     For the single object case, if f_t and h_t are linear functions and the initial state X_1 and noise have a Gaussian distribution, then the optimal state estimate is given by the Kalman Filter.
     - Kalman Filters:
       A Kalman filter is used to estimate the state of a linear system where the state is assumed to be distributed by a Gaussian. Kalman filtering is composed of two steps, prediction and correction. In case the functions f_t and $h_t$are nonlinear, they can be linearized using the Taylor series expansion to obtain the extended Kalman filter. Similar to the Kalman filter, the extended Kalman filter assumes that the state is distributed by a Gaussian.
     - Particle Filters:
       One limitation of the Kalman filter is the assumption that the state variables are normally distributed (Gaussian). The Kalman filter will give poor estimations of state variables that do not follow

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

Gaussian distribution. This limitation can be overcome by using particle filtering. In particle filtering, the conditional state density $p(X_t \mid Z_t)$ at time t is a set of samples $\{s_t^{(n)} : n = 1, \ldots, N\}$ (particles) with weights $\pi_t^{(n)}$ (sampling probability). The weights define the importance of a sample, that is, its observation frequency [17].

- Joint Probability Data Association Filter:
  Let a track be defined as a sequence of measurements that are assumed to originate from the same object. Suppose we have N tracks and, at time t, $Z(t) = z_{1(t)}, \cdots z_{mt}(t)$ are the m measurements. Rasmussen and Hager [16] use a constrained JPDAF filter to track regions.
- Multiple Hypothesis Tracking (MHT):
  If motion correspondence is established using only two frames, there is always a finite chance of an incorrect correspondence. Better tracking results can be obtained if the correspondence decision is not taken until several frames have been examined. The MHT algorithm maintains several correspondence hypotheses for each object at each time frame. MHT makes associations in a deterministic sense and exhaustively enumerates all possible associations.

V. KERNEL TRACKING

Kernel tracking is typically performed by computing the motion of the object, which is represented by a primitive object region, from one frame to the next. The object motion is generally in the form of parametric motion (translation, conformal, affine, etc.) or the dense flow field computed in successive frames.

- Tracking Using Template and Density-Based Appearance Models:
  They have relative simplicity and low computational cost. Two subcategories based on whether the objects are tracked individually or jointly,
  1. Tracking single objects: The most common approach in this category is template matching. Template matching is a brute force method of searching the image, I_w, for a region similar to the object template, O_t defined in the previous frame. The position of the template in the current image is computed by a similarity measure. The stable component identifies the most reliable appearance for motion estimation, that is, the regions of the object whose appearance does not quickly change over time. The transient component identifies the quickly changing pixels. The noise component handles the outliersarised due to noise.
  2. Tracking Multiple Objects: Modelling objects individually does not take into account the interaction between multiple objects and between objects and background during tracking. Isard and MacCormick [18] propose joint modelling of the background and foreground regions for tracking. The background appearance is represented by a mixture of Gaussians. Appearance of all foreground objects is also modelled by mixture of Gaussians. The shapes of objects are modelled as cylinders. They assume the ground plane is known, thus the 3D object positions can be computed. Tracking is achieved by using particle filters where the state vector includes the 3D position, shape and the velocity of all objects. They propose a modified prediction and correction scheme for particle filtering which can increase or decrease the size of the state vector to include or remove objects.
- Tracking Using Multi-view Appearance Models:
  In the previous tracking methods, the appearance models, that is, histograms, templates etc., are usually generated online. The objects may appear different from different views, and if the object view changes dramatically during tracking, the appearance model may no longer be valid, and the object track might be lost. To overcome this problem, different views of the object can be learned offline and used for tracking. Avidan [19] used a Support Vector Machine (SVM) classifier for tracking. SVM is a general classification scheme that, given a set of positive and negative training examples, finds the best separating hyperplane between the two classes. Generally, negative examples consist of background regions that could be confused with the object. Avidan's tracking method maximizes the SVM classification score

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

over image regions in order to estimate the position of the object. One advantage of this approach is that knowledge about background objects is explicitly added in the tracker.

VI. Silhouette Tracking

Objects may have complex shapes, for example, hands, head, and shoulders that cannot be well described by simple geometric shapes. Silhouette-based methods provide an accurate shape description for these objects. The goal of a silhouette-based object tracker is to find the object region in each frame by means of an object model generated using the previous frames. This model can be in the form of a color histogram, object edges. We divide silhouette trackers into two categories, namely, shape matching and contour tracking. Shape matching approaches search for the object silhouette in the current frame. Contour tracking approaches evolve an initial contour to its new position in the current frame by either using the state space models or direct minimization of some energy functional.

- Shape Matching:
  Shape matching can be performed similar to tracking based on template matching; where an object silhouette and its associated model is a searched in the current frame. The search is performed by computing the similarity of the object with the model generated from the hypothesized object silhouette based on previous frame. The silhouette is assumed to only translate from the current frame to the next, thus non-rigid object motion is not explicitly handled. Tracking is achieved by evaluating the optical flow vector computed inside the hypothesized silhouette such that the average flow provides the new object position. Tracking silhouettes can be performed by computing the flow vectors for each pixel inside the silhouette such that the flow is used to generate the silhouette trajectory. Sato and Aggarwal [20] proposed to generate object tracks by applying Hough transform in the velocity space to the object silhouettes in consecutive frames.
- Contour Tracking:
  Contour tracking methods iteratively evolve an initial contour in the previous frame to its new position in the current frame. This contour evolution requires that some part of the object in the current frame overlap with the object region in the previous frame. Tracking by evolving a contour can be performed using two different approaches. The first approach uses state space models to model the contour shape and motion. The second approach directly evolves the contour by minimizing the contour energy.
  - Tracking Using State Space Models:
    The object's state is defined in terms of the shape and the motion parameters of the contour. The state is updated at each time instant such that the contour's a posteriori probability is maximized. The posterior probability depends on the prior state and the current likelihood which is usually defined in terms of the distance of the contour from observed edges. MacCormick and Blake extended the particle filter-based object tracker in Isard and Blake [17] to track multiple objects by including the exclusion principle for handling occlusion.
  - Tracking by Direct Minimization of Contour Energy Functional:
    Both the segmentation and tracking methods minimize the energy functional either by greedy methods or by gradient descent. The contour energy is defined in terms of temporal information in the form of either the temporal gradient (optical flow). Contour tracking using temporal image gradients is motivated by the extensive work on computing the optical flow. The optical flow constraint is derived from the brightness constancy constraint: $I^{t+1}(x, y) - I^{t}(x - u, y - v) = 0$, where I is the image, t is the time, and (u, v) is the flow vector in the x and the y directions. The objective was to compute u and v iteratively for each contour position using the level set representation. Mansouri [21] also uses the optical flow constraint for contour tracking. His approach is motivated by computing the flow vector for each pixel inside the complete object region in a circular neighbourhood with radius r using a brute force search.
    Multiple Camera Tracking:
    The need for using multiple cameras for tracking comes into picture for two reasons. The first reason is the use of depth information for tracking and occlusion resolution. The second reason for using multiple cameras is to increase the area under view since it is not possible for a single camera to observe large areas because of a finite sensor field-of-view. An important issue in using multiple cameras is the relationship between the different camera views which can be manually defined [22].

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

VII. CONCLUSION

Significant progress has been made in object detection and tracking during the last few years. Many robust trackers have been developed which can track objects in real time in simple scenarios. Tracking and associated problems of feature selection, object representation, dynamic shape, and motion estimation are very active areas of research and new solutions are continuously being proposed. One challenge in tracking is to develop algorithms for tracking objects in unconstrained videos. The use of a particular feature set for tracking can also greatly affect the performance. The features that best discriminate between multiple objects and, between the object and background are also best for tracking the object. In this paper, we present an extensive survey of object tracking methods. We divide the tracking methods into three categories based on the use of object representations, namely, methods establishing point correspondence, methods using primitive geometric models, and methods using contour evolution.

REFERENCES

1. Bashir Muhammad and Syed AbdRahman Abu-Bakar, “A hybrid skin color detection using HSV and YCgCr color space for face detection”, Signal and Image Processing Applications (ICSIPA), IEEE International Conference, pp. 95-98, 2015.
2. Juseong Lee, Hoyoung Tang and JongsunPark , “Energy Efficient Canny Edge Detector for Advanced Mobile Vision Applications”, IEEE Transactions on Circuits and Systems for Video Technology, 2016.
3. Raffaele Gaetano, Giuseppe S. and Giovanni Poggi, “Texture-based segmentation of very high resolution remote-sensing images”, 9 th International conference on Intelligent Systems Design and Applications, pp. 578-583, 2009.
4. Moravec , H. , “Visual mapping by a robot rover”, International Joint Conference on Artificial Intelligence (IJCAI), Vol. 2,1979.
5. Harris , C. and Stephens , M., “A combined corner and edge detector”, 4th Alvey Vision Conference, pp. 147–151, 1988.
6. Shi, J. and Tomasi , C., “Good features to track”, IEEE Conference on Computer Vision and Pattern Recognition, pp. 593-600, 1994.
7. Lowe, D. , “Distinctive image features from scale-invariant keypoints”, International Journal of Computer Vision, Vol.60, Issue 2, pp. 91-110, 2004.
8. Mikolajczyk , K., Schmid , C, “A performance evaluation of local descriptors”, European Conference on Computer Vision, Vol.1, pp.128-142, 2003.
9. Wren , C., Azarbayejani , A., and Pentland , A., “Pfinder: Real-time tracking of the human body. from Natural Scene Images: Towards a System for Visually Impaired Persons” , 17th Int. Conference on Pattern Recognition IEEE Transaction Pattern Analysis M*achine Intelligence*, pp. 780–785, vol. 7, 1997
10. Stauffer , C. and Grimson , W, “Learning patterns of activity using real time tracking”, IEEE Transaction Pattern Analysis M*achine Intelligence*., vol.12, Issue 8, pp. 747 – 767, 2000.
11. Comaniciu , D. and Meer , P, “Mean shift: A robust approach toward feature space analysis”, IEEE Transaction Pattern Analysis M*achine Intelligence*, Vol. 24, Issue 5, pp. 603–619, 2002.
12. Shi , J. and Malik , J., “Normalized cuts and image segmentation”, IEEE Trans. Pattern Analysis. Mach. Intell. Volume 22, Issue 8, pp. 888–905, 2000.
13. Viola , P., Jones , M., and Snow , D, “Detecting pedestrians using patterns of motion and appearance”, IEEE International Conference on Computer Vision (ICCV), pp. 734–741, 2003.
14. Yilmaz, A., Javed, O., Shah, M, “Object Tracking: A Survey”, ACM Computing Surveys, Vol. 38, No. 4, 2006.
15. Veenman , C., Reinders , M., and Backer , E., “Resolving motion correspondence for densely moving points”, IEEE Trans. Pattern Analysis M*achine Intelligence*, Vol. 23, No. 1, pp. 54–72, 2001.
16. Rasmussen , C. and Hager , G., “Probabilistic data association methods for tracking complex visual objects”, IEEE Transaction Pattern Analysis M*achine Intelligence*, Vol. 23, No. 6, pp. 560–576, 2001.
17. Isard , M. andBlake , A.,“Condensation - conditional density propagation for visual tracking”. International Journal Computer Vision, Vol. 29, No. 1, pp. 5–28, 1998.
18. Maccormick , J. and Blake , A., “Probabilistic exclusion and partitioned sampling for multiple object tracking”, International Journal Computer Vision, Vol. 39, No.1, pp.57–71, 2000.
19. Avidan , S., “Support vector tracking”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.184–191, 2001.
20. Sato , K. and Aggarwal , J., “Temporal spatio-velocity transform and its application to tracking and interaction”, Computer Vision Image *Understanding*, Vol. 96, No. 2, pp. 100–128, 2004.
21. Mansouri , A., “Region tracking via level set PDEs without motion computation”, IEEE Transaction Pattern Analysis M*achine Intelligence* Vol. 24, No.7, pp. 947–961, 2002.
22. Collins , R., Lipton , A., Fujiyoshi , H., and Kanade , T., “Algorithms for cooperative multisensor surveillance”, Proceedings of IEEE, Vol. 89, No.10, pp. 1456–1477, 2001.

BIOGRAPHY

Subhash Nevhal[1] is an M.E Student in the Computer Engineering Department, Pune Institute of Computer Technology, Pune, Maharashtra, India. His research interests are Computer Vision, Robotics and Machine Learning.

Dr.Archana Ghotkar[2] is an Associate Professor in the Computer Engineering Department, Pune Institute of Computer Technology, Pune, Maharashtra, India. Her research interests are Image Processing and pattern Recognition, Computer Graphics and Visualization, Compiler and Computer Vision.