

VIDEO TRACKING SYSTEM

Archana Hombalimath, Sunitha Sooda, Anisha Kunjan S

Asst Prof, Dept. of CSE, HKBK College of Engineering, vharchana@yahoo.co.in

Asst Prof, Dept. of CSE, HKBK College of Engineering, sunithajagadeesht@gmail.com

Asst Prof, Dept. of ISE, CMRIT College of Engineering, anisha.s@cmrit.ac.in

Abstract- Video tracking is the process of locating a moving object (or multiple objects) over time using a camera. It has a variety of uses, some of which are: human-computer interaction, security and surveillance, video communication and compression, augmented reality, traffic control, medical imaging and video editing. Video tracking can be a time consuming process due to the amount of data that is contained in video. Adding further to the complexity is the possible need to use object recognition techniques for tracking over time. However, video needs more space for storage and wider bandwidth for transmission

Video tracking is a complex problem because the environment, in which video motion needs to be tracked, is widely varied based on the application and poses several constraints on the design and performance of the tracking system. Current datasets that are used to evaluate and compare video motion tracking algorithms use a cumulative performance measure without thoroughly analyzing the effect of these different constraints imposed by the environment. But it needs to analyze these constraints as parameters.

The advance of technology makes video acquisition devices better and less costly, thereby increasing the number of applications that can effectively utilize digital video. Compared to still images, video sequences provide more information about how objects and scenarios changes.

Index term: video, video tracking, image segmentation, tracing system, object detection, compression, object tracking

INTRODUCTION

Object tracking can be defined as the process of segmenting an object of interest from a video scene and keeping track of its motion, orientation, occlusion etc. in order to extract useful information. Object tracking in video processing follows the segmentation step and is more or less equivalent to the 'recognition' step in the image processing. Detection of moving objects in video streams is the first relevant step of information extraction in many computer vision applications.

Object tracking is required in many vision applications such as human-computer interfaces, video communication/compression, road traffic control, security and surveillance systems. Often the goal is to obtain a record of the trajectory of the moving single or multiple targets over time and space, by processing information from distributed sensors. Object tracking in video sequences requires on-line processing of a large amount of data and is time-expensive. Additionally, most of the problems encountered in visual tracking are nonlinear, non-Gaussian, multi-modal or any combination of these. Different techniques are available in the literature for solving tracking tasks in vision and can be divided in general into two groups: *i*) classical applications, where targets do not interact much with each other, *behave independently* such as aircrafts that do not cross their paths, and *ii*) applications in which targets do not behave independently, their identity is not always very well distinguishable. Tracking multiple identical targets has its own challenges when the targets pass close to each other or merge.

RELATED WORK

Many researchers have tried various approaches for object tracking. Nature of the technique used largely depends on the application domain. Some of the research work done in the field of object tracking includes:

1. *A. Gyaourova, C. Kamath, S. and C. Cheung* has studied the block matching technique for object tracking in traffic scenes. A motionless airborne camera is used for video capturing. They have discussed the block matching technique for different resolutions and complexities [1].
2. *Yoav Rosenberg and Michael Werman* explains an object-tracking algorithm using moving camera. The algorithm is based on domain knowledge and motion modeling. Displacement of each point is assigned a discreet probability distribution matrix. Based on the model, image registration step is carried out. The registered image is then compared with the background to track the moving object [2].
3. *A. Turolla, L. Marchesotti and C.S. Regazzoni* discusses the camera model consisting of multiple cameras. They use object features gathered from two or more cameras situated at different locations. These features are then combined for location estimation in video surveillance systems [3].
4. One simple feature based object tracking method is explained by *Yiwei Wang, John Doherty and Robet Van Dyck*. The method first segments the image into foreground and background to find objects of interest. Then four types of features are gathered for each object of interest. Then for each consecutive frames the changes in features are calculated for various possible directions of movement. The one that satisfies certain threshold conditions is selected as the position of the object in the next frame [4].
5. *Çiğdem Eroğlu Erdem and Bülent San* have discussed a feedback-based method for object tracking in presence of occlusions. In this method several performance evaluation measures for tracking are placed in a feedback loop to track nonrigid contours in a video sequence [5].
6. The earliest relevant review on human motion analysis was probably due to Aggarwal et al [6]. It covered various methods used in articulated and elastic non-rigid motion prior to 1994. As for articulated motion, the approaches with or without a prior shape models were described.
7. Cedars and Shah [7] presented an overview of methods for Motion extraction prior to 1995, in which human motion analysis was illustrated as action recognition, recognition of body parts and body configuration estimation on.
8. Aggarwal and Cai gave another survey of human motion analysis [8], which covered the work prior to 1997. Their latest review [9] covering 69 publications was an extension of their workshop paper [10]. The paper provided an overview of various tasks involved in motion analysis of human body prior to 1998. The focuses were on three major areas related to interpreting human motion: (a) motion analysis involving human body parts, (b) tracking moving human from a single view or multiple camera perspectives, and (c) recognizing human activities from image sequences.
9. A similar survey by Gavrilu [11] described the work in human motion analysis prior to 1998. Its emphasis was on discussing various methodologies that were grouped into 2-D approaches with or without explicit shape models and 3-D approaches. It concluded with two main future directions in 3-D tracking and action recognition.
10. Recently, a relevant study by Pentland [12] centered on person identification, surveillance / monitoring, 3-D methods, and smart rooms / perceptual user interfaces to review the state-of-the-art of “looking at people”. The paper was not intended to survey the current work on human motion analysis, but touched on several interesting topics in human motion analysis and its applications.
11. The latest survey of computer vision based human motion capture was presented by Moeslund and Granum [13]. Its focus was on

a general overview based on the taxonomy of system functionalities, viz. initialization, tracking, pose estimation and recognition. It covered the achievements from 1980 into the first half of 2000. In addition, a number of general assumptions used in this research field were identified and suggestions for future research directions were offered.

12. Different techniques are available in the literature for solving tracking problems in vision. Liang Wang, Weiming Hu, Tieniu Tan [14] focused mainly on Monte Carlo techniques (particle filters) because of their power and versatility [15-18]. The Monte Carlo techniques are based on computation of the state posterior density function by samples, and are known under different names: *particle filters (PFs)* [16], *bootstrap methods* [15] or the *condensation algorithm* [11] which was the first variant applied to video processing. The abbreviation condensation stems from conditional density propagation.
13. Color histograms have been widely used for tracking problems [19-21] because they are robust to partial occlusion and are rotation and scale invariant. They have limitations in areas where the background has a similar color as the target object in that the tracker can be confused and lose track of the object. Color histograms also have poor performance when the illumination varies.
14. Other features have been proposed for tracking including shape [22] and gradient direction [23] or a combination of shape and color [24]. A single feature does not provide enough information about the object being tracked and hence using multiple features can provide more information about the object.

OBJECT CLASSIFICATION

Different moving regions may correspond to different moving targets in natural scenes. For instance, the image sequences captured by surveillance cameras mounted in road traffic scenes probably include pedestrians, vehicles, and other moving objects such as flying birds, flowing clouds, etc. To further track people and analyze their activities, it is very necessary to correctly distinguish them from other moving objects.

At present, there are two main categories of approaches towards moving object classification.

Shape-based classification

Different descriptions of shape information of motion regions such as representations of point, box, silhouette and blob are available for classifying moving objects. For example, classified moving object blobs into four classes such as single human, vehicles, human groups and clutter, using a viewpoint-specific three-layer neural network classifier. Input features to the network were a mixture of image-based and scene-based object parameters such as image blob dispersedness, image blob area, apparent aspect ratio of the blob bounding box, and camera zoom. Classification was performed on each blob at every frame, and the results of classification were kept in histogram. At each time step, the most likely class label for the blob was chosen as the final classification.

Motion-based classification

Generally speaking, non-rigid articulated human motion shows a periodic property, so this has been used as a strong cue for moving object classification. For example, a similarity-based technique to detect and analyze periodic motion. By tracking moving object of interest, they computed its self-similarity as it evolved over time. As we know, for periodic motion, its self-similarity measure was also periodic. Therefore they applied time-frequency analysis to detect and characterize the periodic motion, and implemented

tracking and classification of moving objects using periodicity. Optical flow is also very useful for object classification.

Two common approaches mentioned above, namely shape-based classification and motion-based classification can also be effectively combined for moving object classification [15]. Furthermore, Stauffer proposed a novel method based on time co-occurrence matrix to hierarchically classify both objects and behaviors. It is expected that more precise classification results can be obtained by using extra features such as color and velocity.

TYPICAL MOTION AND OBSERVATION MODELS

Motion Models

The techniques used to accomplish a given tracking task depend on the purposes, and in particular on *i*) the objects possessing certain characteristics : cars, people, faces; *ii*) objects possessing certain characteristics with a specific attribute, e.g. moving cars, walking people, talking faces, face of a given person; *iii*) objects of *a priori* unknown nature but of specific interest, such as moving objects. In each case part of the input video frame is searched against a reference model describing the appearance of the object. The reference can be based on image patches, which describe the appearance of the tracked region at the pixel level, or on contours, and/ or on global descriptors such as color models.

To characterize a target, first a feature space is chosen. The *reference object (target) model* is represented by its PDF in the feature space. For example, the reference model can be the color PDF of the target . In the subsequent frame, a *target candidate* is defined at some location and is characterized by the PDF. Both PDFs are estimated from the data and compared by a *similarity function*. The local maxima in the similarity function indicate the presence of objects in the second image frame having a representation similar to the reference model defined in the first frame. Examples of similarity functions are the Bhattacharyya distance and the Kullback-Leibler distance.

In the light of tracking a specified object or region of interest in image sequences, different object models have been proposed in the literature. Many of them make only weak assumption about the precise object configuration and are not particularly restrictive about the types of objects.

Observation models

The observation models for object tracking in video sequences are usually highly nonlinear and can be either *parametric* or *nonparametric* . Some of the most often used observation models are based on color, shape and/ or motion cues. The localization cues impact the tracker based on a PF in different ways. Usually, likelihood models of each cue are constructed. These cues are assumed mutually independent; having in mind that any correlation that may exist between, e.g., the color, motion and sound of an object is likely to be weak. Adaptation of the cues is essential in distinguishing different objects, making tracking robust to appearance variations due to changing illumination and pose etc.

METHODS USED IN THE TRACKING PROCESS

Model-based tracking

Traditionally, the geometric structure of human body can be represented as stick figure, 2-D contour or volumetric model. So body segments can be approximated as lines, 2-D ribbons and 3-D volumes accordingly.

Stick figure

The essence of human motion is typically addressed by the movements of the torso, head and four limbs, so the stick-figure representation can be used to approximate a human body as a combination of line segments linked by joints. The stick figure is obtained in various ways, e.g., by means of median axis transform or distance transform. The motion of joints provides a key to motion estimation and recognition of the whole figure.

2-D contour

This kind of representation of human body is directly relevant to the human body projection in the image plane. In such description, human body segments are analogous to 2-D ribbons or blobs. A silhouette or contour is relatively easy to be extracted from both the model and image. Based upon 2-D contour representation, Niyogi and Abelson used the spatial-temporal pattern in XYT space to track, analyze and recognize walking figures.

Volumetric models

The disadvantage of 2-D models is its restriction to the camera's angle, so many researchers are trying to depict the geometric structure of human body in more detail using some 3-D models such as elliptical cylinders, cones, spheres. The more complex 3-D volumetric models, the better results may be expected but they require more parameters and lead to more expensive computation during the matching process.

An important advantage of 3-D human model is the ability to handle occlusion and obtain more significant data for action analysis. However, it is restricted to impractical assumptions of simplicity regardless of the body kinematics constraints, and has high computational complexity as well.

Region-based tracking

The idea here is to identify a connected region associated with each moving object in an image, and then track it over time using a cross-correlation measure. Region-based tracking approach has been widely used today. The region-based tracking approach works reasonably well. However, difficulties arise in two important situations. The first is that of long shadows. This problem may be resolved to some extent by making use of color or exploiting the fact that shadow regions tend to be devoid of texture. The more serious, and so far intractable, problem for video tracking has been that of congested situations. Under these conditions, people partially occlude one another instead of being spatially isolated. This makes the task of segmenting individual humans very difficult. The resolution to this problem may require tracking systems using multiple cameras.

Feature-based tracking

Abandoning the idea of tracking objects as a whole, this tracking method uses sub-features such as distinguishable points or lines on the object to realize the tracking task. Its benefit is that even in the presence of partial occlusion, some of the sub-features of the tracked objects remain visible. Feature-based tracking includes feature extraction and feature matching. Low-level features such as points are easier to extract. It is relatively more difficult to track higher-level features such as lines and blobs. So, there is usually a trade-off between feature complexity and tracking efficiency.

In addition, Segen and Pingali's tracking system utilized the corner points of moving silhouettes as the features to track, and these

feature points were matched using a distance measure based on positions and curvatures of points between successive frames. The tracking of point and line features based on Kalman filtering has been well developed in the computer vision community. Another tracking aspect, the use of multiple cameras has recently been actively studied. Multi-camera tracking is very helpful for reducing ambiguity, handling occlusions and providing general reliability of data.

For the tracking systems based on multiple cameras, one needs to decide which camera or image to use at each time instant. That is to say, it is an important problem for a successful multi-camera tracking system how the selection and data fusion between cameras are handled.

HOW DOES VISUAL TRACKING WORK?

First, we need a description for the object to be tracked. This can, for example, be a template image of the object, a shape, texture color model or something alike. Building such an initial object description is a very critical and hard task, because the quality of the description directly relates to the quality of the tracking process. Additionally, such a description is not always available to the tracking application beforehand and thus, it may need to be built up during runtime.

Second, objects are usually embedded into certain context. Visual context has been successfully studied in object detection tasks as well as the image understanding field

Most visual tracking methods include image input, appearance feature description, context information integration, decision and model update, as shown in Fig. 1. For different methods, emphasis is not the same, so their schemes will be different. Due to the great success of Particle Filtering, also known as sequential Monte Carlo methods (SMC), visual tracking has been formulated as a problem of Bayesian inference in state space. Compared with the regular exhaustive search-based methods, the main advantage of the use of a particle filter is the reduction of sampling patches during tracking. Another benefit of the particle filter is that the sampling effort can be kept constant, independent a to the size of the object to track which is not the case with simply expanding the search region around the object with fixed factor. Therefore, introducing more advanced Monte Carlo sampling methods would greatly elevate the visual tracking performance.

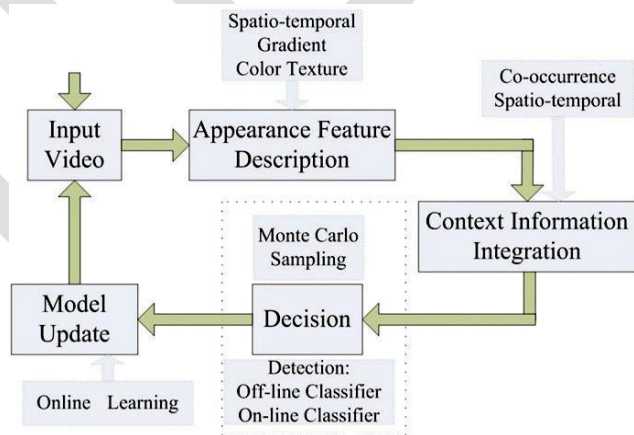
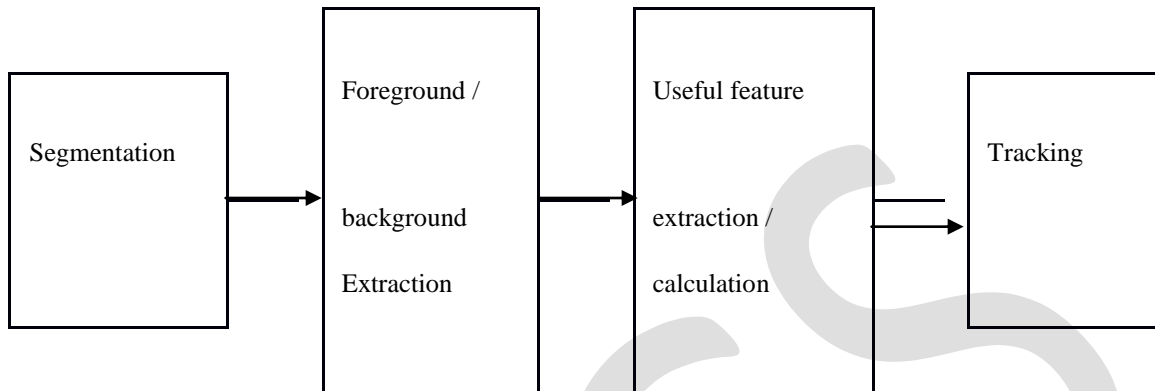


Fig. 1. The flowchart of visual tracking

STEPS IN OBJECT TRACKING

The process of object tracking is summarized in the block diagram below:



Basic steps in object tracking can be listed as:

- Segmentation
- Foreground / background extraction
- Camera modeling
- Feature extraction and tracking

Segmentation

Segmentation is the process of identifying components of the image. Segmentation involves operations such as boundary detection, connected component labeling, thresholding etc. Boundary detection finds out edges in the image. Any differential operator can be used for boundary detection. Thresholding is the process of reducing the grey levels in the image. Many algorithms exist for thresholding.

Foreground extraction

As the name suggests this is the process of separating the foreground and background of the image. Here it is assumed that foreground contains the objects of interest. Some of the methods for foreground extraction are

Use of difference images

In this method we use subtraction of images in order to find objects that are moving and those that are not. The result of the subtraction is viewed as another grey image called *difference image*. Three types of difference images are defined.

- Absolute accumulative difference image is given by
$$f(x,y) = f(x,y) + 1 \dots\dots\dots\text{if } |g(x,y,t_{i+1}) - g(x,y,t_i)| > T$$
- Positive accumulative difference image is given by
$$f(x,y) = f(x,y) + 1 \dots\dots\dots\text{if } g(x,y,t_{i+1}) - g(x,y,t_i) > T$$

- Negative accumulative difference image is given by

$$f(x,y) = f(x,y) + 1 \dots\dots\dots \text{if } g(x,y,t_i) - g(x,y,t_{i+1}) > T$$

The following figures illustrate the three difference images.

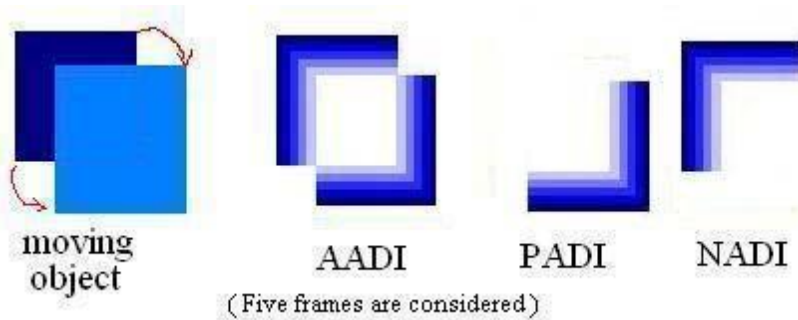
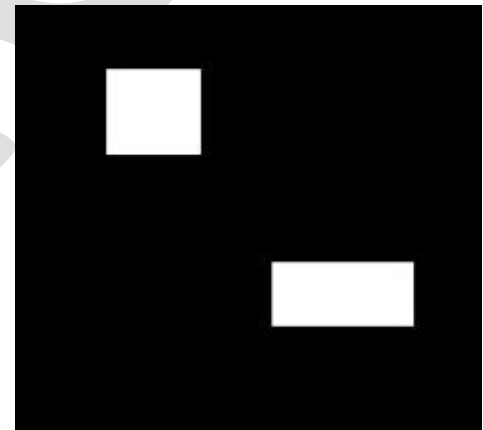


Fig.2

A gap- mountain method described in [4] can then be applied to identify image blocks that are moving and those that are not moving. The gap-mountain method works as follows- Consider a difference image shown in the adjacent figure. A gap is a sequence of consecutive black pixels and mountain is a sequence of consecutive white pixels. If width of a mountain in a particular row is greater than a preset threshold then we assume that a moving object is present in that row. Similar technique is the algorithm proceeds by dividing the image into smaller sub images until each sub matrix contains exactly one object. In the adjacent figure by choosing proper thresholds we can detect the presence of two blocks.



Kalman filtering this method employs 'kalman filter' for predicting the image at t_{i+1} based on some noise model. The difference between predicted and actual intensities is thresholded to classify the image pixel as foreground or background. One advantage of this method is it considers effect of noise, which is very important feature in real world Applications. For example an automatic road traffic management system may detect false objects due to bad weather, wind etc.

Background extraction

Once foreground is extracted a simple subtraction operation can be used to extract the background [67]. Following figure illustrates this operation:

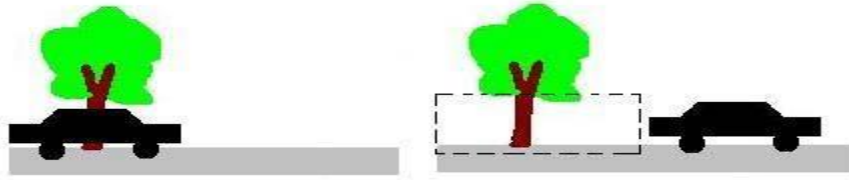


Fig 3. Background extraction

Another method that can be used in object tracking is *Background learning*. This approach can be used when fixed cameras are used for video capturing. In this method, an initial training step is carried out before deploying the system. In the training step the system constantly records the background in order to 'learn' it. Once the training is complete the system has complete information about the background. Though this step is slightly lengthy, it has a very important advantage. Once we know the background, extracting the foreground is matter of simple image subtraction.

Camera modeling

Camera model is an important aspect of any object-tracking algorithm. All the existing objects tracking systems use a preset camera model. In words camera model is directly derived from the domain knowledge. Some of the common camera models are –

1. Single fixed camera

Example: Road traffic tracking system

2. Multiple fixed cameras

Example: Simple surveillance system

3. Single moving camera

Example: Animation and video compression systems

4. Multiple moving cameras

Example: Robot navigation system

For a single fixed camera no extra processing is necessary. In case of multiple cameras, we get inputs from more than one source. Hence first some 3D transformations are required to adjust all the inputs. This what is done in [3]. For a moving camera, we need some heuristic about camera motion. If exact information about the camera movement is available then it can be included in the form of transformations. Having multiple moving cameras is very complicated situation (but can be faced with in many real world Applications). It needs the algorithm to model motion of all the cameras as well as to integrate results from all the cameras. For multi-camera tracking systems, one needs to decide which camera or image to use at each time instant. That is, the coordination and information fusion between cameras are a significant problem.

Feature extraction and object tracking

The next step is to extract useful features from the sequence of frames. Depending on the algorithm, definition of 'feature' may vary. The next few sections explore some of the important techniques used for tracking:

In the feature based approach discussed by Yiwei Wang, John Doherty and Robert Van Dyck [4] four features namely centroid, dispersion, grayscale distribution, object texture are used for tracking objects. The features are defined as follows:

Centroid = (c_x , c_y) where,

$$c_x = \frac{\sum(p_{ij} * i)}{\sum(p_{ij})}$$

$$c_y = \frac{\sum(p_{ij} * j)}{\sum(p_{ij})}$$

$$\text{dispersion} = \frac{\sum \sqrt{((i - c_x)^2 + (j - c_y)^2) * p_{ij}}}{\sum p_{ij}}$$

In current frame under consideration each useful object is assigned the feature vector. Based on the domain knowledge some expected 'tracks' could be generated. Thus the tracking algorithm becomes finding the best track for each object. For this a matrix X of m objects versus n tracks is computed. An element X[i][j] is the number of features that 'match' with the observed features based on some threshold. Further for matrix a threshold for eligible tracks is set (generally 3). If in a row there is only one track satisfying the eligibility threshold then it becomes the current track of the object. The row and column corresponding to the object and the track is removed from the matrix. For objects with multiple possible tracks, weights are given to the features to evaluate 'cost' for each track. The track that gives the least cost is assigned to the object. Here the weights are given purely on the basis of domain knowledge or based on some heuristic about usefulness of the feature.

Block matching method for tracking

The block matching technique [1] gives good results when single fixed camera is used to capture video. The performance degrades considerably in the presence of snowfall or when moving camera is used. The blocks are usually defined by dividing the image frame into non-overlapping square parts. The measure used for matching is Mean Absolute Difference (MAD), which is given by

$$MAD = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n |A(i, j) - B(i, j)|$$

Each block from the current frame is matched into a block in the destination frame by shifting the current block over a predefined neighborhood of pixels in the destination frame. At each shift, the sum of the distances between the gray values of the two blocks is computed. The shift, which gives the smallest total distance, is considered the best match. Other than Mean Absolute Difference (MAD), mean squared distance (MSD), and normalized cross-correlation (NCC) can be used for matching.

Exploiting the domain knowledge

T_ Huang_ D_ Koller_ J_ Malik_ G_ Ogasawara_ B_ Rao_ S_ Russell_ and J_ Weber discussed a different approach in which domain knowledge is exploited to simplify object tracking. As the objects under consideration are vehicles, image path can be

approximated by affine transformations. Since motion is constrained to the road plane and since possible rotation components along the normal of the plane are small the degrees of freedom can be reduced to the extent that we obtain a velocity equation of only a scale parameter s and a displacement vector $u(x)$:

$$u(x) = s(x - x_m) + u_0,$$

Compressed domain object tracking

This tracking method [70] uses compressed domain MPEG video as the source. In the method described by Radhakrishna Achanta, Mohan Kankanhalli, phillipe Mulhem user selects the object of interest. The bounding rectangle of object R_c is then traced in the compressed domain I frame. This region is projected onto the predicted P and B frames. The histogram matching operation is performed to track the object. For histogram matching clipped DCT coefficients for C_b and C_r are used. A measure called diffused sum defined by

$$DiffSum = \sum_{n \in [1,9]} Wt[n] (|HDiffCr| + |HDiffCb|)$$

is used for histogram comparison. Here $HDiffCr$ and $HDiffCb$ are histogram bin differences and $Wt[n]$ is weight factor of particular histogram bin. Higher weights are used for DC and low frequency AC values and lower weights are used for relatively higher AC values.

ARCHITECTURAL AND PERFORMANCE CONSIDERATIONS

Object tracking and other video-processing applications are computation intensive operations; hence performance considerations become critical when they are to be used in real time systems. Following are the various factors affecting the system architecture (i.e. memory, processor, degree of parallelism)

- Whether real time response is required or not
- Is the processing carried out in compressed domain or not
- Affordable budget

A typical approach to evaluating the performance of the detection and tracking system uses ground truth to provide independent and objective data (e.g. classification, location, size) that can be related to the observations extracted from the video sequence. Manual ground truth is conventionally gathered by a human operator who uses a 'point and click' user interface to step through a video sequence and select well-defined points for each moving object. The manual ground truth consists of a set of points that define the trajectory of each object in the video sequence. The human operator decides if objects should be tracked as individuals or classified as a group. The motion detection and tracking algorithm is then run on the pre-recorded video sequence and ground truth and tracking results are compared to assess tracking performance.

The reliability of the video tracking algorithm can be associated with a number of criteria: the frequency and complexity of dynamic

occlusions, the duration of targets behind static occlusions, the distinctiveness of the targets (e.g. if they are all different colors), and changes in illumination or weather conditions.

CONCLUSION

From the discussion, it can be seen that object tracking has many useful applications in the robotics and computer vision fields. Several researchers have explored and implemented different approaches for tracking. The success of a particular approach depends largely on the problem domain. In other words, a method that is successful in robot navigation may not be equally successful in automated surveillance. Further there exists a cost/performance trade off. For real time applications we may need a fast high performance system on the other hand offline applications we may use a relatively cheap (and slower in performance). It can also be seen from the diverse nature of the techniques used that the field has a lot of room for improvement.

REFERENCES:

1. A. Gyaourova, C. Kamath, S. and C. Cheung - Block matching object tracking - LLNL Technical report, October 2003
2. A. Turolla, L. Marchesotti and C.S. Regazzoni – Multicamera object tracking in video surveillance applications
3. Çiğdem Eroğlu Erdem and Bülent San - Video Object Tracking With Feedback of Performance Measures - IEEE Transactions on circuits and systems for video technology, vol. 13, no. 4, April 2003
4. Y. Wang, J. Doherty and R. Van Dyck – Moving object tracking in video - *Proc. Conference on Information Sciences and Systems*, Princeton, NJ, March 2000.
5. R. Achanta, M. Kankanhalli, P. Mulhem - Compressed domain object tracking for automatic indexing of objects in MPEG home video - Proc. IEEE conference in Multimedia and Expo (ICME 2002), Lausanne, August 2002.
6. T.B. Moeslund, E. Granum, A survey of computer vision-based human motion capture, *Computer Vision and Image Understanding*, 81 (3) (2001)
7. C. Cedras, M. Shah, Motion-based recognition: a survey, *Image and Vision Computing*, 13 (2) (1995) 129-155.
8. J.K. Aggarwal, Q. Cai, Human motion analysis: a review. *Proc. of IEEE Workshop on Motion of Non-Rigid and Articulated Objects*. 1997, pp. 90-102.
9. J.K. Aggarwal, Q. Cai, Human motion analysis: a review, *Computer Vision and Image Understanding*, 73 (3) (1999) 428-440.
10. J.K. Aggarwal, Q. Cai, Human motion analysis: a review. *Proc. of IEEE Workshop on Motion of Non-Rigid and Articulated Objects*. 1997, pp. 90-102.
11. D.M. Gavrila, The visual analysis of human movement: a survey, *Computer Vision and Image Understanding*, 73 (1) (1999) 82-98.
12. Alex Pentland, Looking at people: sensing for ubiquitous and wearable computing, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22 (1) (2000) 107-119.
13. T.B. Moeslund, E. Granum, A survey of computer vision-based human motion capture, *Computer Vision and Image Understanding*, 81 (3) (2001) 231-268.
14. Recent Developments in Human Motion Analysis BY Liang Wang, Weiming Hu, Tieniu Tan* *National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, P. R. China, 100080*
15. I. Haritaoglu, D. Harwood, L.S. Davis, W⁴: real-time surveillance of people and their activities, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22 (8) (2000) 809-830.

16. P. Remagnino, T. Tan, K. Baker, Multi-agent visual surveillance of dynamic scenes, *Image and Vision Computing*, 16 (8) (1998) 529-532.
17. C. Maggioni, B. Kammerer, *Gesture Computer: history, design, and applications*. Computer Vision for Human-Machine Interaction. Cambridge Univ. Press, 1998.
18. W. Freeman, C. Weissman, Television control by hand gestures. *Proc. of Intl. Conf. on Automatic Face and Gesture Recognition*. 1995, pp. 179-183.
19. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking," *IEEE Transactions of Signal Processing*, 2002, pp. 174–188.
20. And K. Nummiaro, E. Koller-Meier, and L. V. Gool, "Color features for tracking non-rigid objects," *Special Issue on Visual Surveillance Chinese Journal of Automation*, 2003, pp. 345–355.
21. D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," *Computer Vision and Pattern Recognition*, vol. 2, 2000, pp. 142–149