MULTI-CAMERA CALIBRATION, OBJECT TRACKING AND QUERY GENERATION

Fatih Porikli, Ajay Divakaran

Mitsubishi Electric Research Labs

ABSTRACT

An automatic object tracking and video summarization method for multi-camera systems with a large number of non-overlapping field-of-view cameras is explained. In this framework, video sequences are stored for each object as opposed to storing a sequence for each camera. Object-based representation enables annotation of video segments, and extraction of content semantics for further analysis. We also present a novel solution to the inter-camera color calibration problem. The transitive model function enables effective compensation for lighting changes and radiometric distortions for large-scale systems. After initial calibration, objects are tracked at each camera by background subtraction and mean-shift analysis. The correspondence of objects between different cameras is established by using a Bayesian Belief Network. This framework empowers the user to get a concise response to queries such as "which locations did an object visit on Monday and what did it do there?"

1. INTRODUCTION

The nature of single-camera single-room architecture multi-camera surveillance applications demands automatic and accurate calibration, detection of object of interest, tracking, fusion of multiple modalities to solve inter-camera correspondence problem, easy access and retrieving video data, capability to make semantic query, and effective abstraction of video content. Although several multi-camera setups have been adapted for 3D vision problems, the non-overlapping camera systems have not investigated thoroughly. In [2], a multi-camera system that uses a Bayesian network to combine multiple modalities is proposed. Among these modalities, the epipolar, homography, and landmark information assume any pair of cameras in the system has an overlapping field-of-view. Due to this assumption, it is not applicable to the single-camera single-room architecture.

A major problem of multi-camera systems is the color calibration of cameras. In the past few years, many algorithms were developed to compensate for radiometric mismatches. Most approaches use registered images of a uniformly illuminated color chart of a known reflectance taken under different exposure settings as a reference [1], and estimate the parameters of a brightness transfer function. Often, they assume the function is smooth and polynomial. However, uniform illumination conditions may not be possible outside of a controlled environment.

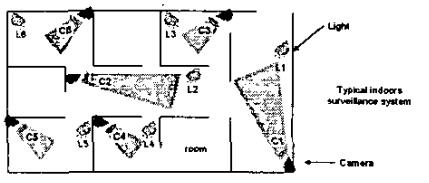

Fig. 1. A multi-camera setup can contain several cameras working under different lighting conditions.

In this paper, we designed a framework where we can extract the object-wise semantics from a non-overlapping field-of-view multi-camera system. This framework has four main components: camera calibration, automatic tracking, inter-camera data fusion, and query generation. We developed an object-based video content labeling method to restructure the camera-oriented videos into object-oriented results. We also propose a summarization technique using the motion activity characteristics of the encoded video segments to provide a solution to the storage and presentation of the immense video data. To solve the calibration problem, we developed a correlation matrix and dynamic programming based method. We use color histograms to determine inter-camera radiometric mismatch. Then, a minimum cost path within the correlation matrix is found using dynamic programming. This path is projected onto diagonal axis to obtain a model function that can transfer one histogram to other. In addition, we use a novel distance metric to determine the object correspondences.

2. RADIOMETRIC CALIBRATION

A typical indoor surveillance system consists of several non-overlapping view of cameras as illustrated in Fig. 1. Usually, such systems consist of identical cameras that are operating under various lighting conditions, or different cameras that have dissimilar radiometric characteristics. Even identical cameras, which have the same optical properties and

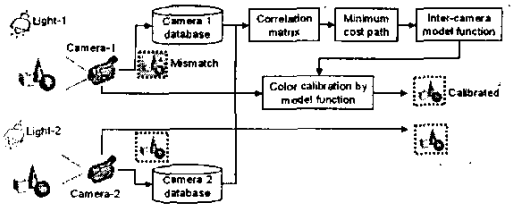

Fig. 2. After computing correlation matrix, a minimum cost path is found and transformed to a model function. Using the model function obtained in the calibration stage, the output of one camera is compensated.

are working under the same lighting conditions, may not match in their color responses. Images of the same object acquired under these variants show color dissimilarities. As a result, the correspondence, recognition, and other related computer vision tasks become more challenging.

We compute pair-wise inter-camera color model functions that transfer the color histogram response of one camera to the other as illustrated in Fig. 2 in the initial calibration stage. First, we record images of the identical objects for each camera. For the images of an object for the current camera pair 1-2, $\mathcal{I}_k^1$,$\mathcal{I}_k^2$, we find color histograms h_k^1, h_k^2. A histogram, h, is a vector $[h[0],\ldots,h[N]]$ in which each bin $h[n]$ contains the number of pixels corresponding to the color range. Using the histograms h_k^1, h_k^2, we compute a correlation matrix $M_k^{1,2}$ between two histograms as the set of positive real numbers that represent the bin-wise histogram distances

$$M^{1,2} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1N} \\ . & & . & . \\ c_{N1} & & \cdots & c_{NN} \end{bmatrix} \quad (1)$$

where each element c_{mn} is a positive real number such that $c_{mn} = d(h_1[m], h_2[n])$ and $d(\cdot) \geq 0$ is a distance norm. Note that, the sum of the diagonal elements represents the bin-by-bin distance with given norm $d(\cdot)$. For example, by choosing the distance norm as L_2 the sum of the diagonals becomes the Euclidean distance of histograms.

An aggregated correlation matrix M is calculated by averaging the corresponding matrices of K image pairs as $M^{1,2} = 1/K \sum_{k=1}^{K} M_k^{1,2}$. Given two histograms and their correlation matrix, the question is what is the best alignment of their shapes and how can the alignment be determined? We reduce the comparison of two histograms to finding the minimum cost path in the matrix $M^{1,2}$. We used dynamic programming and modified Dijkstra's algorithm to find the path $p : \{(m_0, n_0), .., (m_I, n_I)\}$ that has the minimum cost from the c_{11} to c_{NN}, i.e. the sum of the matrix elements on the path p gives the minimum score among all possible routes. We define a cost function for

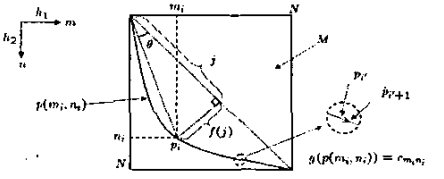

Fig. 3. Relation between the minimum cost path and $f(j)$.

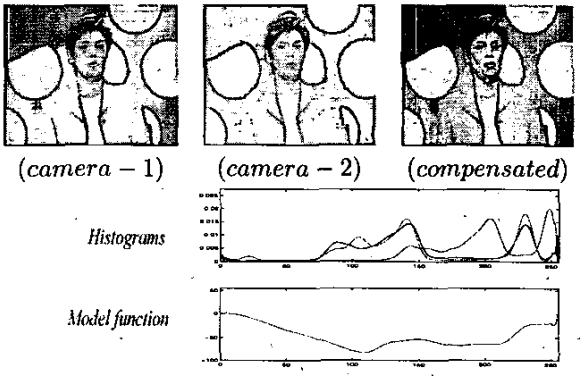

Fig. 4. Camera-2 image is compensated using model func. (Histogram black: camera-1, blue: camera-2, red: compen.)

the path as $g(p_i) = c_{m_i,n_i}$ where p_i denotes the path element (m_i, n_i). We define a mapping $i \rightarrow j$ from the path indices to the projection onto the diagonal of the matrix M, and an associated transfer function $f(j)$ that gives the distance from the diagonal with respect to the projection j. The transfer function is defined as a mapping from the matrix indices to real numbers $(m_i, n_i) \longrightarrow f(j)$ such that $f(j) = |p_i| \sin\theta$ as illustrated in Fig. 3. The correlation distance is the total cost along the transfer function $d_{CD}(h_1, h_2) = \sum_{i=0}^{I} |g(m_i, n_i)|$.

3. SINGLE-CAMERA TRACKING

After initial calibration, objects are detected and tracked at each camera (Fig. 5). A common approach for detecting a moving object for a stationary camera setup is background subtraction. The main idea is to subtract the current image from a reference image that is constructed from the static image pixels during a period of time. We previously developed an object detection and tracking method that integrates a model-based background subtraction with a mean-shift based forward tracking mechanism [3]. Our method constructs a reference image using pixel-wise mixture models, finds changed part of image by background subtraction, removes shadows by analyzing color and spatial properties of pixels, determines objects, and tracks them in the consec-

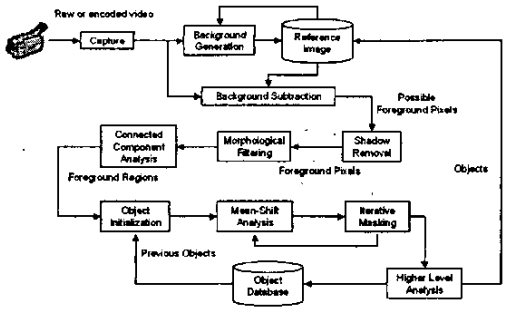

Fig. 5. Single-camera tracking.

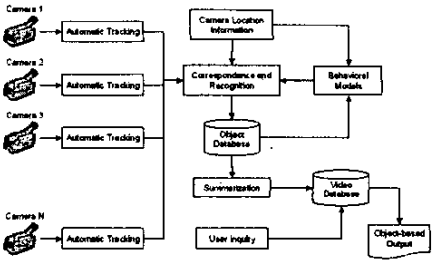

Fig. 6. Multi-camera surveillance system.

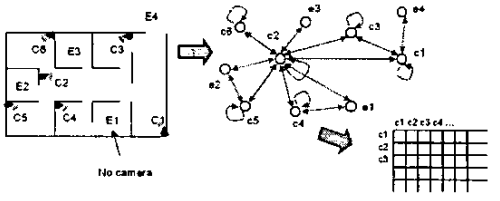

Fig. 7. Each camera corresponds to a node in the directed graph. The links show physical routes between cameras. The probability of object movement is represented using the transition table.

utive frames. In background subtraction, we model the history of a color channel each pixel by a mixture of Gaussian distributions. The reference image is updated by comparing the current pixel with the existing distributions. In case the current pixel's color value is within a certain distance of the mean value of a distribution, the mean value is assigned as the background. After background subtraction, we detect and remove shadow pixels from the set of the foreground pixels. The likelihood of being a shadow pixel is evaluated iteratively by observing the color space attributes and local spatial properties. The next task is to find the separate objects. To accomplish this, we first remove speckle noise, then determine connected regions, and group regions into separate objects. We track objects by computing the highest gradient direction of color histogram, which is implemented as a maximization process. This process is iterated by given the current object histogram extracted from the previous frame.

The tracking results at each camera should be merged to determine the global behaviour of objects (6).

4. INTER-CAMERA CORRESPONDENCE

Another main issue is the integration of the tracking results of each camera to make inter-camera tracking possible. To find the corresponding objects in different cameras and in a central database of the previous appearances, we evaluate the likelihood of possible object matches by fusing the object features such as color, shape, texture, movement, camera layout, ground plane, etc. Color is the most common feature that is widely accepted by object recognition systems since it is relatively robust towards the size and orientation changes. Since adequate level of detail is usually not available in a surveillance video, texture and face features does not improve recognition accuracy. Another concern is the face orientation. Most face-based methods work only for frontal images. The biggest drawback of shape features is the sensitivity to the boundary inaccuracies. Using a height descriptor will only help if we have the ground plane.

There is a strong correlation between camera layout and likelihood of the objects appearing in a certain camera after they exit from another one. As in Fig.7, we formulate the camera system as a Bayesian Belief Network (BBN), which is a graphical representation of a joint probability distribution over a set of random variables. A BBN is a directed graph in which each set of random variable is represented by a node, and directed edges between nodes represent conditional dependencies. In the multi-camera system, each camera corresponds to a node in the directed graph. The links show the physical routes between the cameras. The probability of object movement is the elements of the transition table. The transition probabilities, that is the likelihood of a person moving from the current camera to a linked camera, are learned by observing the system. Note that, each direction on a link may have different probability, and the total incoming and outgoing probability values are equal to one. To satisfy the second constrain, some slack nodes that correspond to the unmonitored entrance/exit regions are added to the graph. Initially, there is no object assigned to any node of the BBN, the number of objects in the cameras and objects in the database are equal to zero. The database keeps track of the individual objects. Let an object O_i is detected at camera C_i. For each detected new object, a database entry is made using its color histogram features. If the object O_1 exits from the camera C_i, then the conditional probability $P_{O_1}(C_j|C_i)$ of the same object will be seen on another cam-

era C_j is found by $P_{O_1}(C_j|C_i) = P(C_j|C_{s1})..P(C_{sk}|C_i)$ where $\{s1, .., sk\}$ is the highest probability path from C_i to C_j on the graph. Because of the dynamic nature of the surveillance system the conditional probabilities change with time, $P_{O_1}(C_j, t|C_i) = P(C_j, t|C_{s1})..P(C_{sk}, t|C_i)$. The conditional probabilities are set to erode by time using a parameter $\alpha < 1$ as $P(O_1, t|C_i) = \alpha P(O_1, t-1|C_i)$ since the object may exit from the system completely (We do not think a multi-camera system should be closed graph). However, the probabilities do not become less than a threshold.

As a new object is detected, it is compared with the objects in the database and with the objects that disappeared previously. The comparison is based on the color histogram similarity and the proposed distance metric. For more than one object correspondence, we select the match as the pair (O_m, O_n) that maximizes $P(O_m, t|C_i)P(O_n, t|C_j)$. We evaluate the matching for all objects simultaneously instead of matching independently to match objects between two cameras C_i and C_j,

5. OBJECT BASED QUERIES

After matching the objects between the cameras, we label each video frame according to the object appearances. This enables us to include content semantics in the subsequent processes. A semantic scene is defined as a collection of shots that are consistent with respect to a certain semantic, in our case, the identities of objects. Since we know camera locations and we extracted which objects appeared in which video at what time, we can, for instance, query for what an object did in a certain time period at a certain location. To obtain concise representation of query results, we generate an abstract of the query result. The key-frame-based summary is a collection of frames that aims to capture all of the visual essence of the video except, of course, the motion. In [4], we showed that the frame at which the cumulative motion activity is half the maximum value is also the halfway point for the cumulative increment in information. Thus, we select the key frames according to the cumulative function.

6. DISCUSSION

A base station controls the fusion of the multi camera information, and its computational load is negligible. We presented sample results of the single-camera tracking module in Fig. 8-a, and the object-based inquiry interface in Fig. 8-b. The tracking method can follow several objects at the same time even some of the objects are totally occluded for a long time. Furthermore, it provides accurate object boundaries. We initialized the Bayesian Network with identical conditional probabilities. These probabilities may also be adapted by observing the long-term motion behaviour of the objects. Sample query results are shown in Fig. 8-c. We are able to extract all appearances of the same person in a specified time period accurately. We can also count the number of different people.

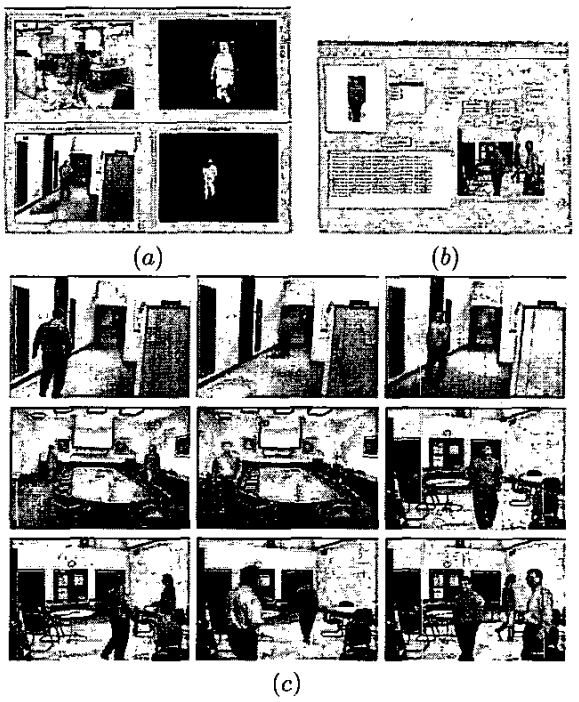

Fig. 8. (a) Inquiry system, (b) extracted instances of an object in all three cameras.

We presented a novel color calibration method. Unlike the existing approaches, our method does not require uniformly illuminated color charts or controlled exposure image sets. Furthermore, our method can model non-linear, non-parametric distortions, and the transitive property makes the calibration of large-scale systems much simpler. The object-based representation enables us to associate content semantics, so we can generate query based summaries. This is an important functionality to retrieve the target video segments from a large database of surveillance video.

7. REFERENCES

[1] M. Grossberg and S. K. Nayar, "What can be known about radiometric resp. func. using images", *Proc. of ECCV*, 2002.

[2] T. Chang and S. Gong. "Bayesian modality fusion for tracking multiple people with a multi-camera system". *AVSS*, 2001

[3] F. Porikli, O. Tuzel. "Human body tracking and movement analysis", *Proc. ICVS-PETS*, Austria, 2003.

[4] A. Divakaran "Video summarization using descriptors of motion activity" *Electronic Imaging*, 2001.