

Color-Based Object Tracking in Multi-Camera Environments

Katja Nummiaro¹, Esther Koller-Meier², Tomáš Svoboda²,
Daniel Roth², and Luc Van Gool^{1,2}

¹ Katholieke Universiteit Leuven, ESAT/PSI-VISICS, Belgium,
{knummiar,vangool}@esat.kuleuven.ac.be

² Swiss Federal Institute of Technology (ETH), D-ITET/BIWI, Switzerland,
{ebmeier,svoboda,vangool}@vision.ee.ethz.ch

Abstract. This paper presents a multi-view tracker, meant to operate in smart rooms that are equipped with multiple cameras. The cameras are assumed to be calibrated³. In particular, we demonstrate a virtual classroom application, where the system automatically selects the camera with the 'best' view on the face of a person moving in the room. Real-time object tracking, which is needed to achieve this, is implemented by means of color-based particle filtering. The use of multiple model histograms for the target (human head) results robust tracking, even when the view on the target changes considerably like from the front to the back. Information is shared between the cameras, which adds robustness to the system. Once one camera has lost the target, it can be reinitialized with the help of the epipolar constraints suggested by the others. Experiments in our research environment corroborate the effectiveness of the approach.

1 Introduction

Intelligent environments like 'smart rooms' pose several research challenges, such as object/people tracking, face/gesture recognition, and speech analysis. There is a rich variety of applications at stake, like surveillance, human-computer interfacing, video conferencing, industrial monitoring, and tele-training.

In this paper we focus on the autonomous processing of visual information based on a network of calibrated cameras. Each camera system comprises a recognition and tracking module which locates the target in the observed scene. Both modules operate on color distributions, where the target model includes a description of the changes of its colors with viewpoint. To document interesting events on-line, an automated virtual editor is included in the central server, which produces a single video stream out of the different camera outputs by systematically selecting the one with the 'best' view. Figure 1 illustrates the system architecture of our multi-camera setup.

For the integration of the multiple cameras, the ViRoom (Visual Room) system by Doubek *et al.* [4] is used. The modular architecture is constructed

³ The full calibration is used, but the mutual epipolar geometry would be sufficient.

from low-cost digital cameras and standard computers running under Linux. It also supports consistent, synchronized image acquisition. The ViRoom has fixed cameras, whereas our smart room has cameras which are mounted on pan-tilt heads. Therefore, we have extended the ViRoom software with automatic camera control in order to keep the target in the center of view.

Currently, the research area of multi-camera systems is very active [2, 7, 8, 12]. For instance, a flexible multi-camera system for low bandwidth communication is presented by Comaniciu *et al.* [2]. Based on color tracking the target on the current image can be transmitted in real-time with high resolution. Khan *et al.* [7] describe an interesting approach to track people with multiple cameras that are uncalibrated. When a person enters the field of view of one camera, the system searches for a corresponding target in all other cameras by using previously compiled field of view lines. Krumm *et al.* [8] describe an approach for tracking in an intelligent environment, using two calibrated stereo cameras that provide both depth and color information. Each measurement from a camera is transformed into a common world coordinate system and submitted to a central tracking module.

The work most closely related to ours is that of Trivedi *et al.* [12], where an overall system specification for an intelligent room is given. The 3D tracking module operates with multiple cameras and maintains a Kalman filter for each object in the scene. In comparison, we use a more general representation of the probability distribution of the object state which allows to initialize this distribution along the epipolar lines when an object enters the field of view of a camera. In our system the best view is selected according to the quality of the tracking results for the individual cameras while Trivedi *et al.* utilize the motion of the tracked target. We also introduce the use of multiple target histogram based on color distributions in tracking.

The outline of this paper is as follows. Section 2 presents a short review of the color-based tracking technique. In Section 3 the multiple target models used for the tracking are explained. Section 4 presents the exchange of information in the camera network while Section 5 explains the selection of the optimal camera view. In Section 6 some experimental results are presented and finally, Section 7 concludes the paper.

2 Tracking

Robust real-time tracking of non-rigid objects is a challenging task. Color histograms provide an efficient feature for this kind of tracking problems as they are robust to partial occlusion, are rotation and scale invariant and computationally efficient. The fusion of such color distributions with particle filters provides an efficient and robust tracker in case of clutter and occlusion. Particle filters [5] can namely represent non-linear problems and non-Gaussian densities by propagating multiple hypotheses simultaneously.

The color-based particle filter [9, 10] approximates the posterior density by a set of weighted random samples $\{(\mathbf{s}_t^{(n)}, \pi_t^{(n)})\}_{n=1}^N$ conditioned on the past observations. Each sample \mathbf{s} represents one hypothetical state of the object, with

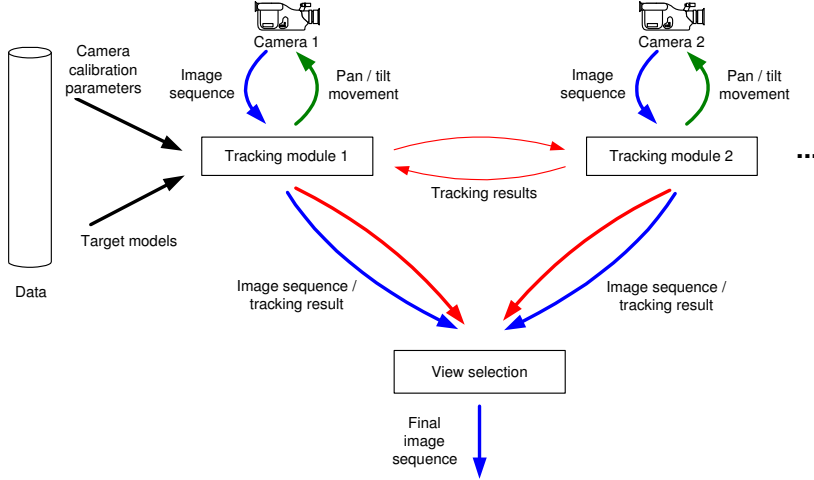


Fig. 1. The sketch of the system architecture with multiple cameras.

a corresponding discrete sampling probability π , where $\sum_{n=1}^N \pi^{(n)} = 1$. The tracked object state is specified by an elliptical region

$$\mathbf{s} = \{x, y, \dot{x}, \dot{y}, H_x, H_y, \dot{H}\} \quad (1)$$

where x, y represent the location of the ellipse, \dot{x}, \dot{y} the motion, H_x, H_y the length of the half axes and \dot{H} the corresponding scale change.

In order to compare the histogram of such a hypothesized region $p_{\mathbf{s}^{(n)}}$ with the target histogram q from an initial model, a similarity measure based on the Bhattacharyya coefficient [1, 3, 6]

$$\rho[p_{\mathbf{s}_t^{(n)}}, q] = \sum_{u=1}^m \sqrt{p_{\mathbf{s}_t^{(n)}}^{(u)} q^{(u)}} \quad (2)$$

is used, where m represents the number of bins for the histograms.

3 Multiple Target Models

To support multiple cameras, a color adjustment (that was already part of the ViRoom software [4]) is applied during the calibration. Another consideration when working with multiple cameras, is using more than one histogram for the target model. For example, when a person walks around in the smart room, camera may at one time have the face in its field of view, and later the back of the head. The color distributions will probably be quite different in these cases, and this correspondence can be established with the multiple cameras.

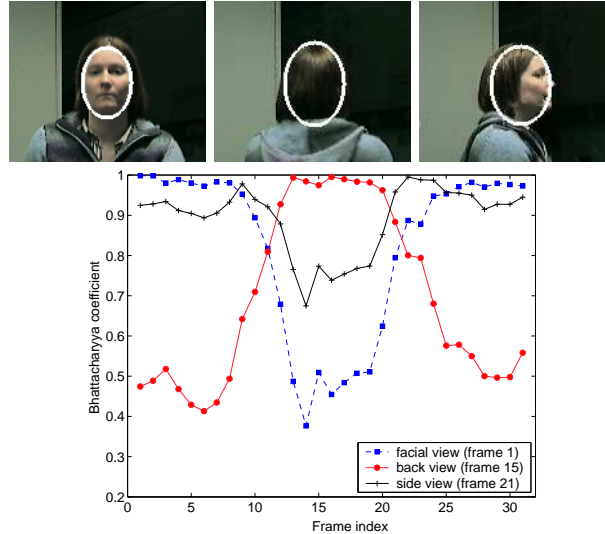


Fig. 2. Top row: The three main views and histogram regions for the target model (frames 1, 15 and 21). Bottom row: The plotted Bhattacharyya values from a 360° turning head sequence, using the different target models (facial, side and back view) and comparing them to the facial region (frame 1).

Three characteristic head images, one from the front, one from the side and one from the back are selected as initial target models and the corresponding histograms $q = \{q_f, q_s, q_b\}$ are stored. During the tracking, the similarity measures to these three histograms are included in the object state. By using a linear stochastic model for the propagation, the Bhattacharyya coefficients for the next frames can be estimated. Rapid changes of these coefficients are therefore avoided. Fig. 2 shows the evolution of the Bhattacharyya coefficients with respect to the three characteristic views shown in the top row as the head gradually turns around. As can be seen, the Bhattacharyya coefficients change smoothly with the viewing angle.

The initial samples of the particle filter for each camera are spread over the whole image or are strategically placed at positions where the target is expected to appear in case such knowledge is available. A target is recognized on the basis of the three Bhattacharyya coefficients, where the best matching model is taken as the target model. By calculating the mean value μ and the standard deviation σ of the Bhattacharyya coefficient for elliptic regions over all the positions of the background in the initialization step, we define the appearance condition as

$$\rho[p_{s_t^{(n)}}, q] > \mu + 2\sigma. \quad (3)$$

This indicates a 95% confidence that a sample does not belong to the background. If a fraction $b \cdot N$ of the samples shows a high enough correspondence to one of the

target histograms the object is considered to be found and the tracking process is started. The parameter $b = 0.1$ has been proven sufficient in our experiments and is called the ‘kick-off fraction’. During tracking, the target model of each camera is adapted as described in our earlier work [9].

4 Exchanging Information across Cameras

Exchanging information across the cameras is important to increase the robustness of the tracking. Such information exchange can take different forms. For instance, cameras could come to a consensus about which (different) side of a face they also see and ensure that these sides are consistent. We are currently implementing this aspect. Here we describe another type of collaboration, that has to do with the area where the targets are expected in the different views, i.e. with the (re)initialization of the individual trackers. In case one or a few cameras have lost the target, the other cameras can provide strong indications where to look based on their epipolar geometry. Epipolar geometry is used in two different ways: 1) During initialization when one of the target models matches an object, 2) When the object is temporally lost due to clutter, occlusions or other difficult tracking conditions.

Initialization: When an object is detected in one camera, we try to initialize it in the other cameras. For this purpose, the epipolar lines are calculated that correspond to the estimated target location. Samples are then placed stochastically around these lines and the velocity components are chosen from Gaussian distributions. If the object is already visible in more than one camera, the samples are distributed around the intersection of the corresponding epipolar lines.

Reinitialization: If less than $b \cdot N$ of the samples fulfill the appearance condition which is explained in Eq. 3, we consider the object to be lost. In this case, we use the epipolar lines and their intersections to reinitialize an object during tracking. A fraction of the samples are then spread around the epipolar lines of the other cameras while the remaining samples are propagated normally.

5 Best View Selection

Faces of people are among the most important targets to be tracked in smart rooms. For applications, where images are transmitted to another location, it will also be important to make an appropriate selection among the views that are available. Accordingly, we have developed an automated virtual editor which creates a video stream by switching to the best view on the face of a person who is freely walking around. Camera hand-over is controlled on the basis of the tracking results by evaluating the Bhattacharyya coefficient (see Eq. 2) of the mean state of each tracker

$$\rho[p_{E[S]}, q_f] = \sum_{u=1}^m \sqrt{p_{E[S]}^{(u)} q_f^{(u)}} \quad (4)$$

$$E[S] = \sum_{n=1}^N \pi^{(n)} \mathbf{s}^{(n)}. \quad (5)$$



Fig. 3. The best view selection according to the Bhattacharyya coefficients of the individual trackers is shown. The small images at the bottom show the tracking results of the individual camera trackers as white ellipses. The numbers represent the corresponding Bhattacharyya coefficients. In the top row the used target model and the output of the virtual editor are displayed.

As the Bhattacharyya coefficient represents a similarity measure with respect to the target histogram, the virtual editor always chooses the camera view which provides the highest Bhattacharyya coefficient for the face histogram (the characteristic frontal view). Figure 3 illustrates the best view selection on the basis of the individual Bhattacharyya coefficients.

6 Results

In this section the experimental results demonstrate the capabilities and limitations of our distributed tracking system. All images are captured from live video streams and have a size of 160×120 pixels. The application runs at 5-8 frames per seconds — without any special optimization — on Pentium III PCs at 1GHz under Linux where each of the three cameras are attached to their own computer. The capability of the virtual editor is illustrated in Figure 4. It can be seen that the camera hand-over automatically chooses the best front view of the tracked face even if it is partly occluded.

The initialization plays an important role in the multi-camera tracker. When there are several potential targets in the neighborhood of the epipolar lines, a wrong object can be selected. Such a scene is shown in Figure 5. In the top row, the situation is still handled correctly as the target is occluded in the middle camera and not initialized. In the second row, the target is not localized correctly whereas in the third row a wrong target is selected. In both cases, the target is occluded in two cameras, so that the samples are spread along an epipolar line and not around an intersection point. Robustness can certainly be increased with more cameras. On the other hand, applications with crowds in the fields of view will also then pose critical problems.



Fig. 4. The virtual editor automatically chooses the best front view of the tracked face.



Fig. 5. The initialization step can cause problems in tracking, if there are several equally good candidates in the vicinity of the epipolar lines.

7 Conclusion

As cameras get cheaper, PCs more powerful, and information traffic more congested, the interest in technology that better supports virtual meetings increases. This paper proposed a multi-camera setup that actively tracks a person. As cameras may see different parts of the head at different times, these changes are supported for through a target model that contains a choice of different color histograms, each corresponding to an interval of viewing angles. The tracker of each camera dynamically chooses the model that matches best. These choices are fed into a ‘virtual editor’, that selects the camera yielding the best view of the face.

Information exchange between the individual trackers is currently only used for the (re)initialization process by applying epipolar geometry. We will push this integration further by implementing a multi-view tracker that combines the geo-

metric and photometric information coming from all cameras in order to process a single, 3D state rather than individual image-by-image states. Furthermore, we will enhance the virtual editor. The camera selection should generate a video stream that is maximally informative and pleasant to watch, avoiding too many short cuts. In addition, the Bhattacharyya coefficient will be combined with alternative decision rules. We also plan to interpolate between available views, in order to create a virtual camera. A meeting setup with multiple people is another interesting extension. The virtual editor should be able to locate the person who is the center of attention at any given time.

Acknowledgment

The authors gratefully acknowledge support by the European Commission project STAR (IST-2000-28764) and the NCCR project IM2, funded by the Swiss National Science Foundation SNF. We thank Petr Doubek and Stefaan De Roeck for the multi-camera set-up, and Bart Vanluyten and Stijn Wuyts for including the active cameras.

References

1. F. Aherne, N. Thacker and P. Rockett, *The Bhattacharyya Metric as an Absolute Similarity Measure for Frequency Coded Data*, *Kybernetika*, pp. 1-7, Vol. 32(4), 1997.
2. D. Comaniciu, F. Berton and V. Ramesh, *Adaptive Resolution System for Distributed Surveillance*, *Real-Time Imaging*, pp. 427-437, Vol. 8, 2002.
3. D. Comaniciu, V. Ramesh and P. Meer, *Real-Time Tracking of Non-Rigid Objects using Mean Shift*, *CVPR*, pp. 142-149, Vol. 2, 2000.
4. P. Doubek, T. Svoboda and L. Van Gool, *Monkeys - a Software Architecture for ViRoom - Low-Cost Multicamera System*, *ICVS*, pp. 386-395, 2003.
5. M. Isard and A. Blake, *CONDENSATION - Conditional Density Propagation for Visual Tracking*, *International Journal on Computer Vision*, pp. 5-28, Vol. 1(29), 1998.
6. T. Kailath, *The Divergence and Bhattacharyya Distance Measures in Signal Selection*, *IEEE Transactions on Communication Technology*, COM-15(1) pp. 52-60, 1967.
7. S. Kahn, O. Javed and M. Shah, *Tracking in Uncalibrated Cameras with Overlapping Field of View*, *PETS*, 2001.
8. J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale and S. Shafer, *Multi-Camera Multi-Person Tracking for EasyLiving*, *International Workshop on Visual Surveillance*, pp. 3-10, 2000.
9. K. Nummiaro, E. Koller-Meier and L. Van Gool, *An Adaptive Color-Based Particle Filter*, *Journal of Image and Vision Computing*, pp. 99-110, Vol 21(1), 2003.
10. P. Pérez, C. Hue, J. Vermaak and M. Gangnet, *Color-Based Probabilistic Tracking*, *ECCV*, pp. 661-675, 2002.
11. T. Svoboda, H. Hug and L. Van Gool, *ViRoom - Low Cost Synchronised Multicamera System and its Self-Calibration*, *DAGM*, pp. 515-522, 2002.
12. M.M. Trivedi, I. Mikic and S.K. Bhonsle, *Active Camera Networks and Semantic Event Databases for Intelligent Environments* *Proceedings of the IEEE Workshop on Human Modelling, Analysis and Synthesis*, 2000.