# Content-based retrieval of visual information

Oerlemans, A.A.J.

# Chapter 10

# Hybrid Maximum Likelihood Similarity

In this chapter we present an object tracking system which allows interactive user feedback to improve the accuracy of the tracking process in real-time video. In addition, we describe the hybrid maximum likelihood similarity, which integrates traditional metrics with the maximum likelihood estimated metric. The hybrid similarity measure is used to improve the dynamic relevance feedback process between the human user and the objects detected by our system.

## 10.1 Introduction

Human Computer Interaction (HCI) will require the computer to have similar sensory capabilities as humans including face [7] [36] [77] [79] and body [11] [27] [35] [40] [54] [79] [81] [96] understanding.

This chapter presents an interactive video tracking system which includes real-time user feedback in both motion detection and object tracking. The feedback from the user is applied in real-time, so the change in the tracking results is immediately visible.

Tracking and identifying moving objects in images from a stationary camera, such as people walking by or cars driving through a scene, has gained much attention in the recent years. It can be a very useful technique for human-computer interaction, the next-generation games, and for surveillance applications [96].

We developed an object tracking system [58] that can analyze live input streams from any video input device and which is able to output the locations, unique identifiers and pictorial information of the moving objects in the scene. All components are plug-ins, so in theory any method for object segmentation, tracking and user feedback can be inserted.

Object detection and identification however is a topic that has its unique set of problems that still are not fully addressed. Multiple object tracking is complicated by the fact that objects can touch or interact with each other, can occlude another, even leave the scene and come back again. And the usual problems with single object tracking, like illumination changes, partial occlusion or object deformation, still apply as well.

To get improved object tracking results, we investigate methods to include user feedback for detecting moving regions and to ignore or focus on specific tracked objects. Our object tracking framework includes color-based relevance feedback [67] functionality at both the segmentation and tracking level.

However, we have seen from earlier experiments [78] that matching the users feedback to new input from the segmentation and tracking algorithms using standard visual similarity metrics, performs poorly in some situations. Similarity metrics that adjust to the true visual similarity are needed. Especially for the constantly slightly changing object appearances, differences in color-feature values do not always have the same visual difference, so we are investigating new similarity metrics that are applied to the object tracking framework, but are applicable in general visual similarity matching as well.

## 10.2 Related work

There has been significant research on motion tracking - an extensive review has been written by Yilmaz et al. [96], that gives a clear overview of object tracking and the challenges and limitations that current object tracking systems face. Notable scientific meetings on object tracking and related research include the Performance and Evaluation of Tracking and Surveillance PETS) and Video Surveillance & Sensor Networks (VSSN).

Relevance feedback [67] is an interactive method that has been successfully introduced into text and image queries. It is an interactive query process where a computers internal representation of the object that a user is interested in is continually updated by providing the system with information about the relevancy of the query outcome.

## 10.3 Visual similarity

### 10.3.1 The maximum likelihood training problem

In chapter 7 of this thesis, the Multi-Dimensional Maximum Likelihood (MDML) paradigm was presented as a method for determining similarity of feature values using the 2D distribution of a training set of these feature values. It was shown that the multi-dimensional approach outperforms the 1D maximum likelihood similarity measure.

In general it is difficult to find sufficient training examples to arrive at a statistically representative model of the underlying probability density function. This fundamental training problem motivates the Hybrid Maximum Likelihood Similarity Measure described next.

### 10.3.2 Hybrid maximum likelihood similarity

In practice, the $L_2$ distance measure is typically a rough but not perfect fit to the underlying similarity measure. Therefore, we propose the Hybrid Maximum Likelihood Similarity (HMLS) measure which interpolates between the $L_2$ distance and the maximum likelihood distance to both obtain a better similarity measure and address the training problem from Section 10.3.1. At both the pixel and object-level feedback, the general algorithm for using the hybrid maximum likelihood similarity measure for a color feature is:

- For each feature vector element x:
  - Initialize a histogram $H_x$ to $H_x[i][j] = (i - j) * (i - j)$
  - Normalize $H_x$ so that the sum of all $H_x[i][j]$ is 1
- When calculating the similarity between to elements at position $x$ with values $i$ and $j$:
  - Use $H_x[i][j]$
- After feedback from the user:
  - Create a new histogram $H_{temp}[i][j]$ for each feature vector element
  - Fill $H_{temp}$ with the feature values from the examples from the user
  - Normalize $H_{temp}$
  - Set $H_x = w * H_x + (1 - w) * H_{temp}$

In this algorithm, $i$ and $j$ range over the possible values of the feature vector element, in our case $[0 \ldots 255]$. The last step in the algorithm generates a new version of the histogram that will converge to the true similarity distribution if enough training samples are given.

## 10.4 Relevance feedback in object tracking

Figure 10.1 gives an overview of our object tracking system and the location of the relevance feedback module in it.

For our first experiments, we decided to use color-based relevance feedback, so we have used our own adaptation of a color-based motion detection method developed by Horprasert et al. [27]. The main idea of their method is to decompose the

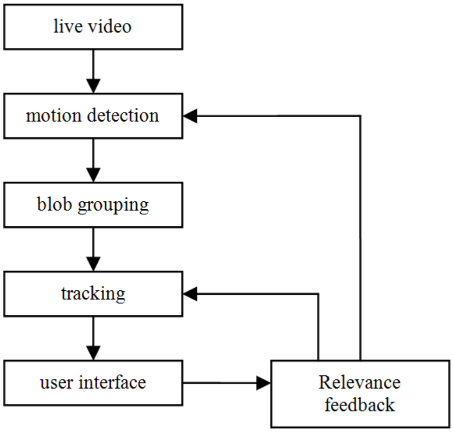

Figure 10.1: The components of the object tracking system with relevance feedback.

difference between the current color information and the modeled background for every pixel into a chromaticity (color) component and a brightness component.

As mentioned in chapter 9, one assumption that the authors of this motion detection method made, was that the lighting would stay roughly constant. In real world applications however, the light can change gradually. Thus, we implemented an adaptive version of their model to compensate for dynamic real world lighting. Small parts of the background model are continuously updated with new data, to compensate for these gradual changes in lighting. Another effect of this adaptation is that deposited objects can be added to the background model if they do not move for a given period of time. For further details, please refer to section 9.3.2 of chapter 9.

We use bounding boxes as a method for object tracking. Objects are considered either simple or compound. Simple objects are objects that can be tracked by the straightforward version of the object tracker, in which every blob corresponds to no more than one object. In the case of object interactions, or overlapping objects, there is ambiguity as to which blob belongs to which object. We define compound objects as virtual objects which consist of two or more objects which are currently interacting in some fashion. The object tracker will track these objects just as simple objects, but will also track the actual simple objects which are involved.

### 10.4.1 Pixel-level feedback

Our object tracking system continuously compares pixel values to the modeled background to decide whether a pixel should be considered as part of a moving object. The relevance feedback component can change this decision by learning from user input. An example is given in figure 10.2. In this case, the user indicates that pixels that look like the selected part of the image (the brick wall) should never be considered to be an object, even if the background model indicates that it should, which could happen in case of fast lighting changes.

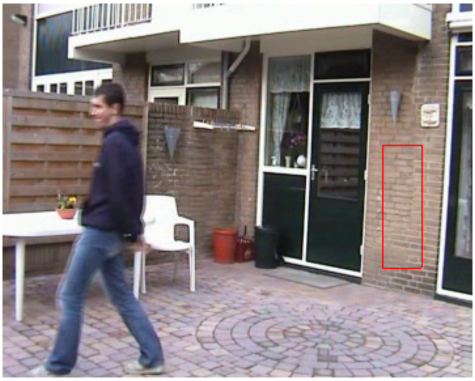

Figure 10.2: User feedback for the object segmentation algorithm: selecting a negative example.

The user can supply feedback while the tracking system is running. The selected positive and negative examples are immediately included in the decision process, so the effect on the object tracking results is instantly visible.

The HMLS is trained using all pairs of pixels in the area that the user has selected.

### 10.4.2 Object-level feedback

We will now present an example of using feedback for the tracking algorithm. Figure 10.3 shows a frame from a sequence in which a person is leaving an object behind. In figure 10.4, the user selects a positive example for the object tracking algorithm. In this case, objects with similar color will always remain marked as foreground and they will not be added to the background model, which is the normal behavior for adaptive object tracking algorithms. Figure 10.5 shows the object being classified as a foreground object based on the user feedback.

Negative examples for the object tracking algorithm are useful for marking objects that are not interesting to the user. The tracking algorithm will ignore objects that are similar to the examples supplied by the user.

The HMLS metric is trained using information on the tracked object from each frame in which it is still tracked.

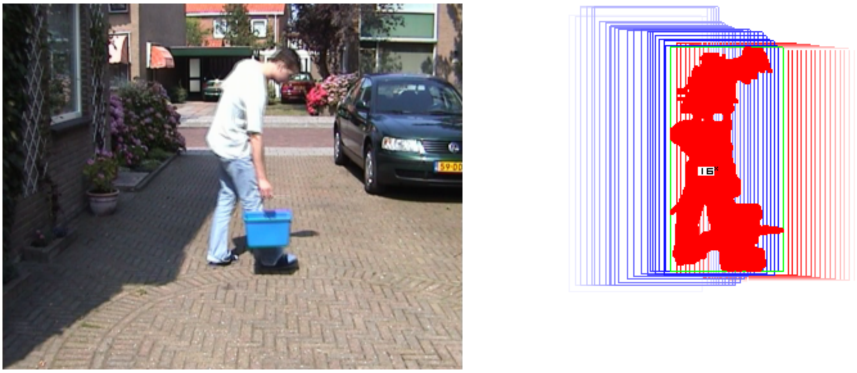

Figure 10.3: One frame from a sequence where someone leaves an object behind, together with the object tracking results.

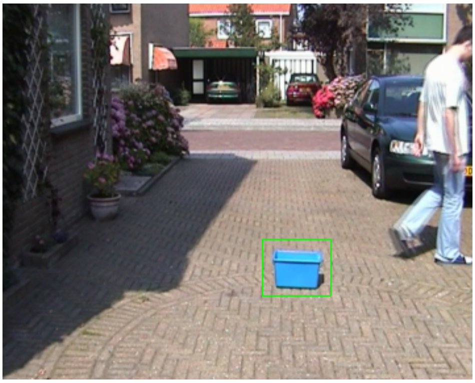

Figure 10.4: The user selects a positive example for the object tracking algorithm.

## 10.5 Conclusions and future work

Based on user surveys, our interactive video tracking system has shown that including relevance feedback in the motion detection and object tracking process is intuitive and promising.

The strong point of the HMLS is that it gives the benefits of maximum likelihood similarity estimation while also addressing the limited training set problem.

In future work, we are interested in treating the temporal space as a wavelet based texture [73], learning optimal features [38], and performing more extensive quantitative evaluation including comparing different similarity measures.

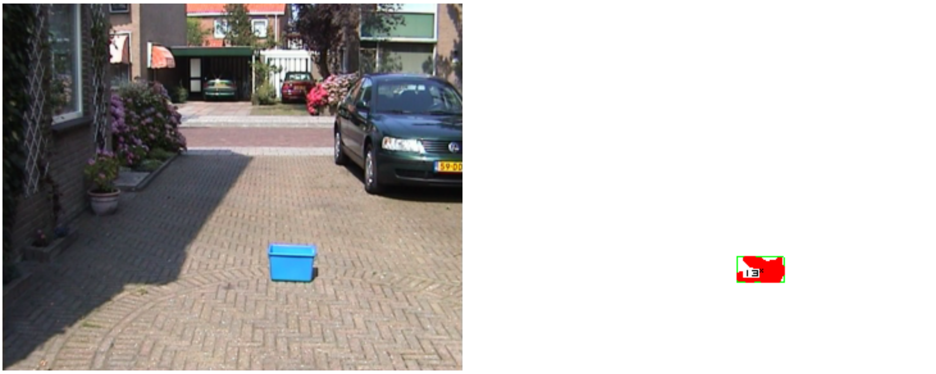

Figure 10.5: The object tracking using the positive example. The object will not be added to the background model and stays visible as a tracked object.