Motion Tracking Project Synopsis

Fabian Wauthier
University of California, Berkeley
Email: flw@berkeley.edu

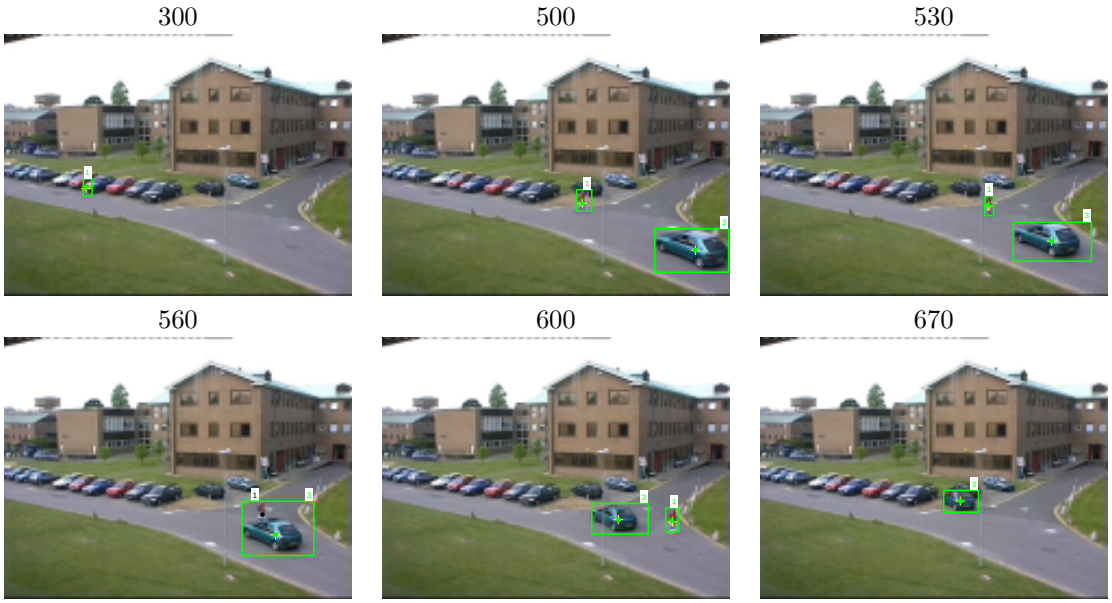

Figure 1: Tracking objects through a video sequence. Tracked objects are shown at different frames of a sequence and highlighted by labelled bounding boxes.

Introduction

This fourth year honours project was concerned with the task of *motion tracking* in video sequences. Given an image sequence from a stationary camera, it was desired to detect and track foreground objects (for instance cars or people) moving through the scene as exemplified in Figure 1. Based on work by Stauffer and Grimson [3] a tracking algorithm was implemented and key claims verified. Algorithm dynamics were analysed and extensions to their work proposed and evaluated. Additionally, work towards an appearance-based object model based on Connor and Reid [2] was undertaken.

Synopsis

The motion tracking task was decomposed into two independent subproblems. The first is to detect foreground objects on a frame-wise basis, by labelling each pixel in an image frame as either foreground or background. The second is to couple object observations at different points in a sequence to yield the object's motion trajectory.

Foreground detection

Initial parts of the project are based on work by Stauffer and Grimson [3] which computes the foreground/background pixel labellings for each sequence image by identifying pixels that are sufficiently unusual in the context of their previous values. To accomplish this, a distinct background model that captures the distribution of recent pixel values (in our case a Mixture of isotropic Gaussians) is maintained for each pixel. As the sequence proceeds each model's statistics (here the mixing proportions, means and variances of the Gaussians) are updated through a set of rules that progressively push probability mass towards recently observed pixel values, thereby absorbing new evidence. This is accomplished either by a set of three recursive update equations that modify a single Gaussian's statistics based on its current parameters, or by replacing a Gaussian with one centred at the current pixel. Upon observing a new image, the algorithm compares each

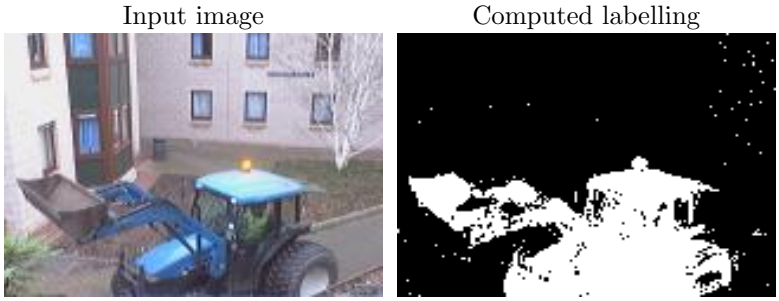

Figure 2: Given a history of previously seen images, the algorithm determined the binary foreground/background labelling seen on the right for the input image on the left. Processing was done on images of size 120×160 pixels.

pixel to its accumulated background model to determine foreground/background membership by a method resembling outlier detection. An example input image and the computed labelling is shown in Figure 2. As can be seen, foreground objects can be extracted from a binary labelling as large connected components, also called *blobs*.

After implementing Stauffer and Grimson's basic algorithm it was desired to understand some of its key properties both by modification and analysis. In order to aid understanding of central dynamics, such as the recursive adaptation of mixture statistics, graphical views were developed that facilitated manual inspection. When running the algorithm on a new sequence, a complex interplay between algorithm, parameters, and the sequence statistics emerges. In order to discover what role relevant design decisions of the algorithm play in this process, a number of modifications were proposed and evaluated. For instance, the updates rules in [3] specify that under certain circumstances a mixture component should be replaced entirely, but it was not clarified precisely which one. Three replacement strategies were evaluated by optimising their associated parameters on a ground truth sequence with respect to a segmentation loss function. The function captures to what degree the computed labelling agrees with the ground truth data. Multiple optimisations were carried out from quasi-random initial parameters, and the final performances of all three techniques compared. Similar experiments were carried out to estimate the benefit of using more flexible mixture components (diagonal instead of spherical Gaussians) in conjunction with appropriate update equations. The analyses indicated no immediate algorithmic improvements, however, the optimisation technique that was developed proved useful for parameter tuning on novel sequences.

The previous experiments illuminated concrete design decisions made by Stauffer and Grimson, but fail to foster an intuitive understanding of the recursive update equations for Gaussian mixture components. Under the assumption of a Gaussian stochastic process, and working in the expectation domain, the recursions were unrolled and rewritten in closed-form. This showed that in expectation each recursion estimates a component statistic (i.e. the mixing proportion, the mean or the variance of the Gaussian) as an exponentially weighted average between an initial estimate and the true process statistic. In order to describe the algorithm dynamics, Stauffer and Grimson provide a rough estimate of a time constant in terms of parameters to the equations, but give no further discussion. Through the foregoing analysis the precise time constant was derived and, using a Taylor expansion, was found to be well approximated by the author's estimate for reasonable parameter ranges. These derivations allowed a natural interpretation of key parameters as determining concrete time constants that control the algorithm's dynamics.

The previous notions were made precise by a case study of the evolution of a single pixel's background model while being exposed to gradual illumination changes and the passing of a foreground object. Three representative input frames are shown in the top row of Figure 3 where the analysed pixel is highlighted by a red box. The bottom row illustrates the evolution of the three mixture

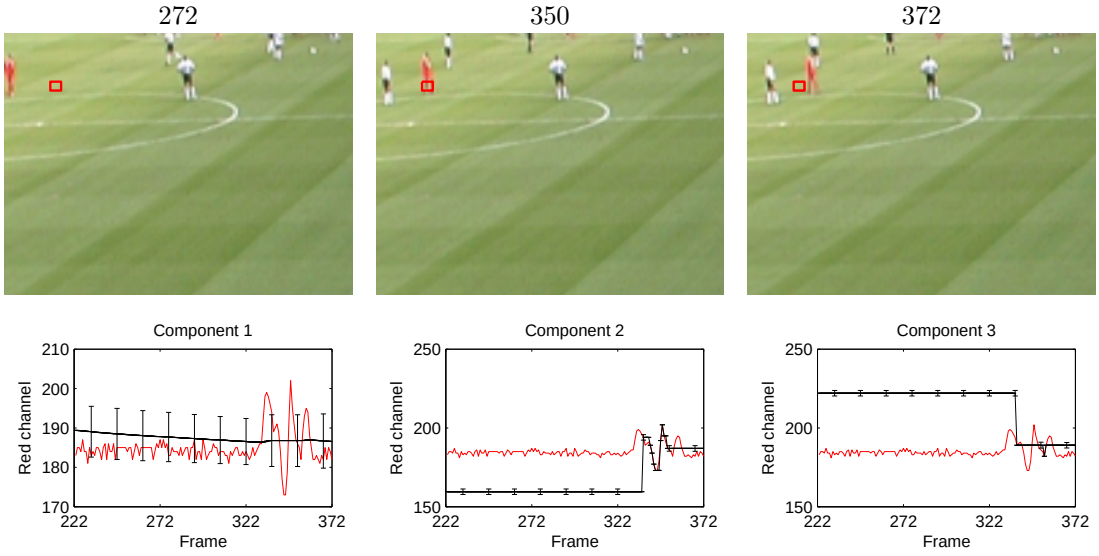

Figure 3: The evolution of a Mixture of Gaussians, comprised of three mixture components. The upper row shows three example frames of a sequence, and the lower row the evolution of three mixture components during that sequence. Shown in red is the value of the red channel of the input sequence, and in black the corresponding mean estimate with error bars indicating one standard deviation.

components for frames 222 through 372. Using visualisations such as these, the previously analysed dynamics were made concrete for a particular sequence. Stauffer and Grimson's central claim is that the algorithm is robust against changes in scene geometry and periods of increased noise while adequately detecting foreground objects. To verify their claim the algorithm was run on a live video stream for 24 hours, some results of which can be seen in Figure 4. The experiment largely confirmed Stauffer and Grimson's claims but also revealed subtle deficiencies of the basic algorithm. Experiments also indicate that the independence assumption encoded by diagonal Gaussians is inappropriate to describe the typical variability observed in RGB data of natural background scenes (lighting changes lead to highly correlated colour components) and highlighted the use of alternative colour spaces.

Object tracking

Using the computed labelling, foreground objects can be identified as large connected components in a binary labelling map. However, direct use of these blobs for tracking is sensitive to noise and fails to determine object identities at different times in a sequence. In order to couple a sequence of *state* measurements, that is, relevant blob properties such as the centroid, centroid velocity between consecutive frames, or size into a denoised state trajectory, Stauffer and Grimson used Kalman filtering. The filter operates as a two-stage process switching between absorbing evidence from the most recent foreground blob and forming a predictive distribution for the next. In the context of tracking multiple objects simultaneously, the key difficulty lies in determining which Kalman filter should absorb which new state measurement. If this association task is not solved adequately then Kalman filters are corrupted with wrong information. The presentation of Stauffer and Grimson was imprecise regarding the employed strategy, thus the second part of the project was mainly concerned with developing a suitable matching framework.

Our approach is encapsulated in Figure 5 where, based on observed dynamics in previous frames, the true identities of objects in frame $t + 1$ must be extrapolated. More specifically, the data

Figure 4: Representative frames showing various computed foreground labellings during day and night-time in a 24 hour experiment. The system adapts to varying lighting levels and changing scene geometry (note the parked car at time 18:53:50 that was correctly labelled as background after a few minutes).

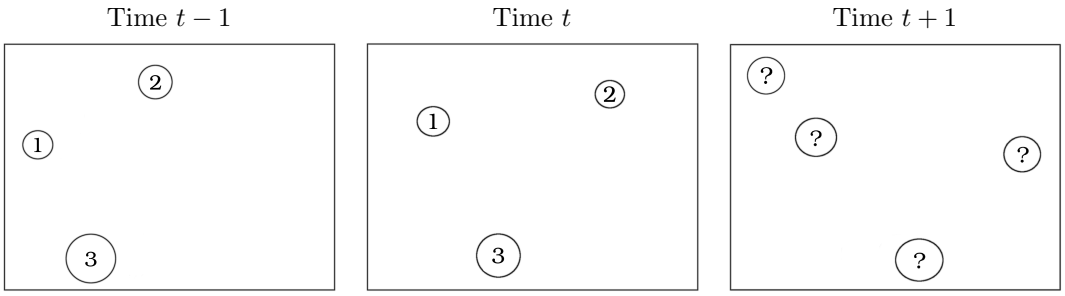

Figure 5: A schematic representation of the data association problem. The object identities at time $t+1$ must be extrapolated while being robust against missing data and noise.

association problem is viewed as finding a perfect matching between blobs and Kalman filters which optimises an objective function that is the sum of all pairwise matching costs (it is thus often called the linear assignment problem, or LAP). Mathematically, a set of pairwise costs was represented by square cost matrix E for which a permutation matrix U, $u_{ij} \in \{0, 1\}$ was desired so that the summed cost

$$c(E, U) = \sum_{ij} e_{ij} u_{ij}$$

is minimal. The cost function was chosen to ensure that an optimal assignment maximises the joint likelihood of all pairings while respecting the various matching constraints. As the cost function depends on the chosen object state representation, experiments were carried out to demonstrate encodings that can inform various association tasks.

Existing algorithms that solve the Linear Assignment Problem, such as the implementation of Jonker and Volgenant's method [1] which was used, compute it as a matching between two sets of same size. This is problematic when the number of available observations does not equal the number of Kalman filters (i.e. noisy observations or temporary object occlusions). Additionally, since a computed matching must necessarily be one-to-one, the tracker would occasionally be forced to accept unlikely matchings (consider the case when actual object measurements are absent and only detection noise is available). To solve these problems, two transformations of a cost matrix E were developed. Firstly, to deal with spurious or absent measurements, a rectangular cost matrix R can be augmented to a square matrix E by extending with a constant, but otherwise arbitrary dummy value ε_d. It was shown that the permutation matrix U that solves the Linear Assignment Problem on E directly induces a minimal-cost matching for the rectangular problem R for any constant ε_d. Secondly, to relax the one-to-one matching constraints an $n \times n$ cost matrix E can be further extended to a $2n \times 2n$ matrix F by adding three $n \times n$ blocks of dummy values ε_d. It was shown that the permutation matrix V that solves the Linear Assignment Problem on F induces pairwise matching costs not exceeding ε_d. The parameter ε_d is thus the maximum acceptable matching cost, thereby allowing us to relax the matching constraints on the original problem R. By combining these two methods the data association problem was phrased in terms of readily available algorithms while appropriately dealing with temporary object occlusions and additional noise.

Evaluations

To complete the project, the full algorithm was evaluated on a number of different sequences, exposing both its strengths and weaknesses. As previous experiments indicated, foreground detection can produce useful information under varying conditions, especially when data is recoded in a different colour space. Manual inspection of various tracking results suggested that multiple

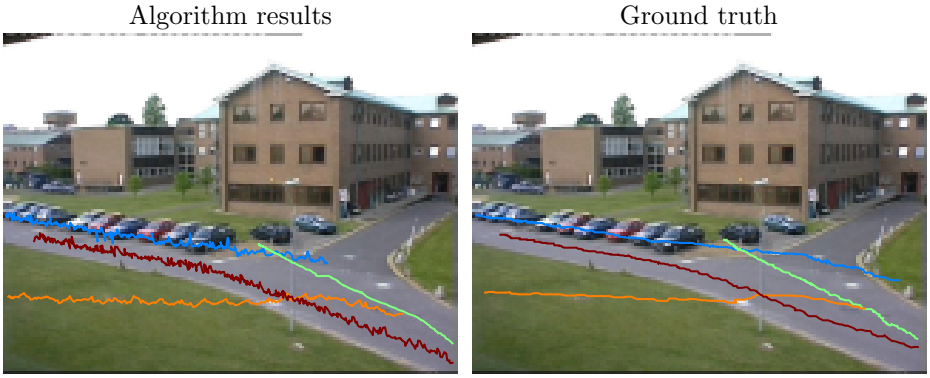

Figure 6: A summary of select motion trajectories. Shown on the left are computed trajectories, and on the right the manually labelled ground truth. A broad agreement can be observed.

object tracking using the devised association technique performs adequately in many situations. Example tracking snapshots for one sequence were previously presented in Figure 1. A summary of some those tracking results is compared against manually labelled ground truth data in Figure 6. Despite the overall success, the algorithm frequently fails to track multiple objects when one is briefly occluded by another, due to the limited representational power of foreground blobs.

Further work

The failure to track multiple objects that interact by occlusion highlights the blob tracking framework as fundamentally inappropriate for reasoning about occlusions. It is more useful to think about these interactions in terms of their causes (namely an object occluding another) rather than unpicking a complex mixture of their effects (i.e. several blobs merging into one). To address this limitation, a generative model for image formation was adapted from Reid and Connor [2] and added to the tracker. The model maintains a *sprite* for each object that describes its appearance (i.e. a "cardboard cut-out" of the object) and can explain the formation of an observed image by appropriately occluding a background image with various objects sprites. It is hoped that by building a more complete model of how objects interact and occlude in the real world, future tracking improvements will be facilitated.

References

[1] R. Jonker and A. Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38(4):325–340, 1987.

[2] I.D. Reid and K.R. Connor. Multiview segmentation and tracking of dynamic occluding layers. In *BMVC 2005*, 2005.

[3] Chris Stauffer and W. Eric L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–757, 2000.