Interactive Object Segmentation in Video by Fitting Splines to Graph Cuts

Iddo Drori Tommer Leyvand Daniel Cohen-Or Hezy Yeshurun

School of Computer Science *
Tel Aviv University

Object segmentation in image sequences is one of the fundamental problems in computer vision and graphics. This problem is usually addressed either by discrete representations which are currently manifested by graph partitioning techniques, or by continuous methods typically referred to as active contours. In this work we take a unified approach by fitting splines to graph cuts. The strengths of this approach stem from the dual discrete and continuous representations and from allowing the user to refine the result of the cut by fitting a new spline to it and modifying its points. Segmentation of an object in video is performed by a series of updates to the control points and computation of a minimum graph cut. Usually the graph cut results in a discrete representation over which the user has no control, and which is not always our desired result. Therefore our approach is to fit a spline to the resulting cut in key-frames. This allows the user to change the control points of the spline and then perform additional iterations of cut computation.

In order to extract an object from an input sequence the user first marks a point on the object and an area around it in a single frame. The object position is tracked forwards and backwards in time and the result is a set of discrete points (x,y,t). We fit a 3D spline to this set of points by a greedy algorithm. First, all of the discrete points are part of the spline. Next, in each iteration, we find the point whose removal from the set results in a new spline minimizing the overall error between the tracker data and spline. The error is global and measures the distance between the spline and the collection of points. Fitting is performed until reaching a minimum number of key-frames. The result is refined by the user by adding key-frames and modifying positions.

We use discrete and continuous representations for both object position and shape. After tracking the object position in each frame and fitting a 3D spline, the user roughly marks the object shape in a number of key-frames by points on a 2D closed spline. The last updated key-frame shape is propagated forward in time as a starting point for the following frames which the user selects to modify. The user can refine key-frame shapes by modifying spline points and applying rigid transformations. Shapes in the remaining frames are initially interpolated from key-frames. Next, the 2D shape is refined by iterative graph cuts.

Graph partitioning techniques are commonly used for object segmentation. We perform dilation on the spline boundaries and compute the minimum graph cut in the boundary region. Voxels on the inner boundary of the dilated region are connected to a source node and voxels on the outer boundary are connected to a sink. The rest of the voxels in the dilated region are connected regularly. The weight of each internal edge is inversely proportional to the gradient magnitude so that the resulting cut sticks to edges and transition regions in the video. We compute a spline approximation to the discrete set of data points resulting from the cut computation. We use a cubic B-spline representation and solve a constrained minimization problem iteratively. The knots of the spline are located automatically, given a single parameter which controls the tradeoff between the smoothness and closeness of fit. A small value results in an accurate spline with many knots, whereas a large value results in a smooth weighted least-squares polynomial with a few knots. Fitting is performed by computing successive least-squares splines. At each iteration computation of spline knots is adaptive, adding more in regions in which the function is difficult to approximate than where it is smooth. This gives a continuous representation which the user can further refine by modifying the spline and then re-computing a new graph cut.

The process of interleaving a discrete cut computation and spline fitting is performed until the user is satisfied with the mask. We have found this method of interaction to be simple and efficient, producing a detailed mask using relatively few steps, benefiting from the advantages of both representations. We demonstrate the result of the process for an underwater image sequence by segmenting a fish. A frame of the input is shown in (a). The projection of the tracker data and the 3D spline fitted to the data are shown in (b). A user specified key-frame and the result of the iterative graph cut are shown in (c). A new spline is fitted to the graph cut as shown in (d) and modified. The resulting mask is shown in (e) and a composite on a green screen is shown in (f). The accompanying video shows the entire sequence. Our approach to object segmentation has useful applications in video compositing, matting and video completion.

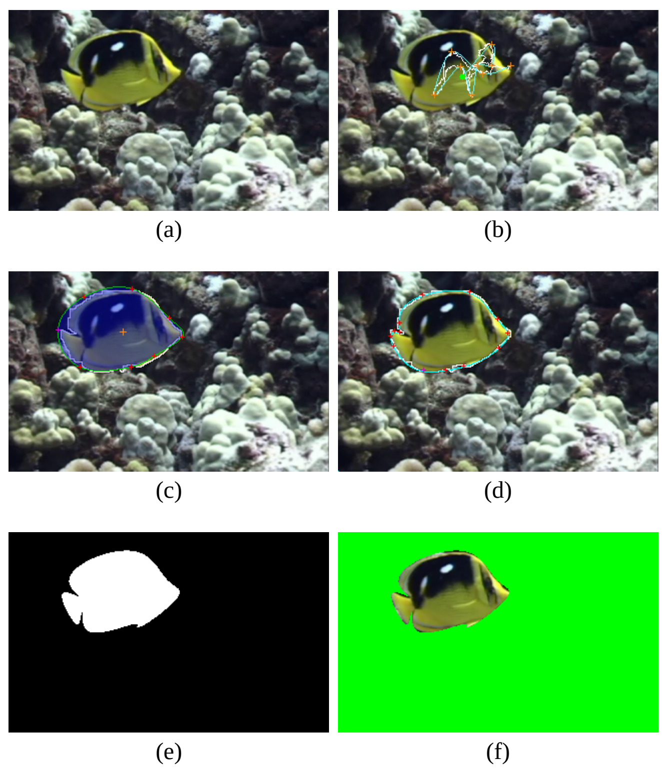

Figure 1: Object segmentation in video: (a) Input frame. (b) Projected tracker data and 3D spline. (c) Key-frame interpolation and minimum graph cut. (d) Spline fitting to graph cut. (e) Binary mask. (f) Composite on green screen.

*e-mail: {idrori | tommer | dcor | hezy}@tau.ac.il