Action Recognition by Shape Matching to Key Frames

Stefan Carlsson Josephine Sullivan

Numerical Analysis and Computing Science
Royal Institute of Technology (KTH)
S - 100 44 Stockholm, Sweden

Abstract

Human action is, in general, characterized by a sequence of specific body postures. An action is often recognizable from a single view of an action specific posture. In this paper we demonstrate that specific actions can be recognized in long video sequence by matching shape information extracted from individual frames to stored prototypes representing key frames of the action. The matching algorithm is tolerant to substantial deformation between image and prototype and recognizes qualitatively similar image shapes produced by the body postures. The algorithm is applied to the recognition of specific tennis strokes in up to 2 min long video sequences. Using key-frames within the sequence all (10-15) forehand strokes are found with no false positives. The backhand strokes are detected with somewhat lower performance. We conclude that this type of approach to action recognition has large potential for application to automatic video analysis and editing.

1. Introduction

The recognition of human activity in a video sequence is a problem that has been approached from many directions. In general it is considered as quite complex, involving all the problems that plague static objects like figure ground segmentation and viewpoint invariance. In addition, action recognition has to deal with the fact that objects are moving non rigidly. Many attempts to do action recognition have therefore involved some kind of 3D modelling of the motion over time [7, 13, 16, 19, 20].

If view invariance is not an issue, which is often the case for many applications, recognition can be based directly on the image data. An important issue, then, is to decide to what extent dynamic information should be exploited. By using all the images in a sequence, motion information is, of course, exploited implicitly e.g. [1, 11, 12], but as pointed out in [5] this in general requires segmentation of the active person in the image. At least for static cameras, this can be avoided by basing the recognition directly on the estimated image motion [3, 4, 5, 18, 22].

Given an image action sequence, ideally all frames in the sequence should be exploited for recognition of the action. However, frequently human actions can be recognized from single frames and we suggest an algorithm for doing this. These specific frames will be called action specific key-frames. The single frame recognition can, of course, be applied to more than one frame in an action sequence thereby increasing the robustness of the recognition. By demonstrating that action can be recognized from single frames, we have a more robust way of exploiting the information in an action image sequence.

A certain human action can be divided into a sequence of poses. Given just one frame in such a sequence, humans can often visually identify the type of action. A person, in general, repeats his sequence of poses with some slight variation when the action is performed at two different time instances. It can be expected that the pose variation between two different persons performing the same action is in general larger than the variation for a single person over time. In this paper, we will consider the recognition of a specific action of a specific person over a time span of 2 min. Specifically, forehand and backhand strokes of tennis players are identified. For each person in each sequence one key-frame per stroke is defined. This key-frame is selected from a typical posture associated with the forehand or backhand stroke. Using this key-frame other frames in the sequence, depicting the same action dependent posture, are found.

2 Shape Matching

When the action is performed at two different time instances, there may be considerable differing posture variation from the key-frame. This variation reflects the changes of the projected image shape of the body. The algorithm presented here is based on the estimation of the deformation of the image shape relative to the shape of the key-frame. The shape is represented in the form of edge data from the Canny-edge detector.

The idea of measuring shape deformation relative to pro-

totypes has a long history in pattern recognition [8] , biology [6]and computer vision [21]. Lately it has been realized by different groups that deformation analysis has to be preceeded by some mechanism for establishing correspondence between shapes. The computation of correspondence between point sets is the issue of the work in [14]. The work in [10] and [2] presents an algorithm for computing correspondence between arbitrary shapes.

Deformation and correspondence can also be captured by the invariants of the deformation. For a specific equivalence class of shapes, like the projections of an action specific key-frame, invariants, correspondence and deformations will be specific and interlinked. The invariants play a crucial role as if they can be defined, they will allow us to compute correspondences and hence estimate the deformation.

The metric properties of the projected shape for the same posture at two different time instants will not have any simple relationship. This can be seen from the examples of extracted shapes of a tennis playing sequence in figure 1. These frames are projected from approximately the same part of the action cycle at different time instants and are overlaid on the action specific key frame by manually adjusting heads and shoulders to fit. We see that there is a substantial deformation still remaining. Ther is also a substantial difference due to missing and added edges. However, if

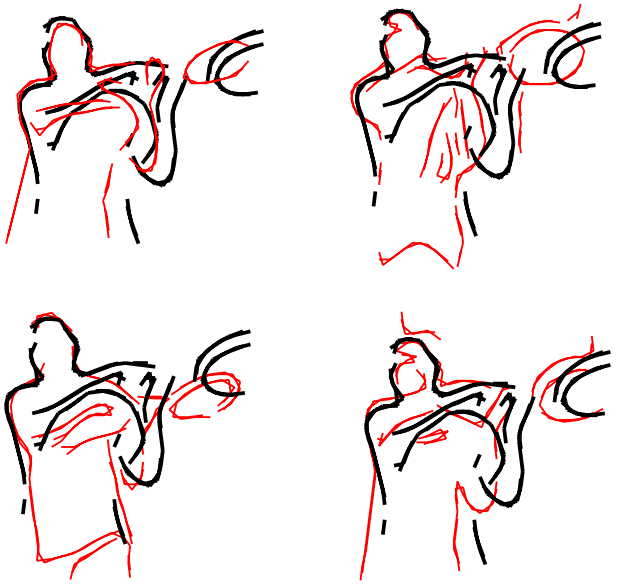

Figure 1: Even after manual alignment of key frame to fit heads and shoulders, a substantial deformation remains relative to sequence frames projected from the same point of the action cycle

we consider qualitative properties such as relative position and directions of different body parts they will in general be invariant. Qualitative geometric relationships can be described in an efficient way using the concept of *order structure* [15]. T

his concept generalises the concept of ordering from one to several dimensions. For a point set, order structure is defined as the set of orientations, clockwise or anti-clockwise, that can be assigned to any triplet as it is traversed sequentially. Order structure for arbitary number of points can be defined bt considering triplet subsets. The combination of points and lines allows us to compute a topological type index based on the relative orientation of the point (x_j, y_j) and line (a_i, b_i) which can be computed as:

$$sign(a_i x_j + b_i y_j + 1) \quad i \neq j$$

For a group of points and lines these signs define the topological type index which caracterizes the group in a qualitative manner. Any small perturbation of the points or lines will leave the topological type invariant. It can, therefore, be used for matching qualitatively similar geometric structures. This was done in [9, 10] and we will use a computationally improved version of the algorithm presented in [10].

This method starts with extraction of edge data. The edge points and their associated tangent lines are sampled and the resulting point-line set is used to represent the image. In order to characterize the order structure for this set we consider all combinations of four points and their associated tangent lines. For each such four-combination we compute the topological type which can be represented by a number, see figure 2 The topological type can be used to compute point to point correspondence between two different shapes via a voting matrix. This matrix's entries are updated by one for corresponding points in the four-point combinations whenever they have the same topological type (see figure 3). After selecting all combinations of four points in images A and B we obtain a matrix of votes between all points in image A and all points in image B. Ideally corresponding points will have high votes. A simple way to select unique correspondences is to apply the Greedy algorithm on this voting matrix. The maximum vote in the matrix is selected and correspondence declared between the points for that vote. The row and collumn of this entry are deleted and the process is repeated while the matrix is non-empty.

An example of the output of the matching algorithm can be seen in figure 4. It displays two frames with approximately the same pose and the computed correspondence. The images contain 137 and 176 edge points respectively, For display purposes only the 100 strongest matches are shown. A vast majority of the matches are correct but some errors can be seen, notably in the tennis racket which was cut off in one of the images.

The algorithm as described works essentially as the geometric hashing algorithm described in [17] with affine struc-

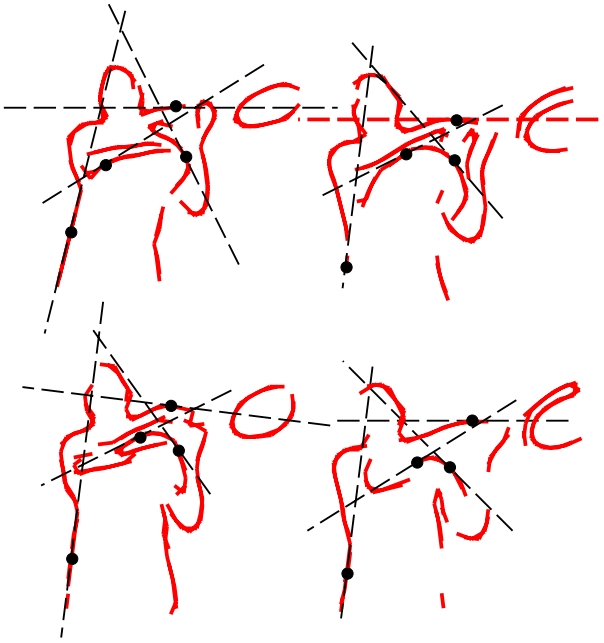

Figure 2: Four points and lines with the same topological type from four different shapes.

ture replaced by topological type. It should be noted however that in geometric hashing, quantized bins have to be defined explicitly, which in a sense determines the equivalence classes of shapes. In our algorithm hashing bins are defined automatically since topological type, in contrast to affine shape, is a discrete entity and can be assigned directly to an index.

3. Efficient Computation of the Voting Matrix

One of the key issues in an algorithm like this is complexity. Since we choose combinations of four points the complexity will grow as n^4, where n is the number of edge points. Compared to [10] we have substantially improved the implementation speed, by noting that the voting matrix can be computed as the product of two matrices. These two matrices represent histograms of topological types for the points in the two images to be matched. The algorithm works by assigning a topological type index to each *ordered* sequence of four points in the images A and B:

$$\begin{aligned} a_1 \quad a_2 \quad a_3 \quad a_4 \quad &\longrightarrow \quad index(a_1, a_2, a_3, a_4) \\ b_1 \quad b_2 \quad b_3 \quad b_4 \quad &\longrightarrow \quad index(b_1, b_2, b_3, b_4) \end{aligned} \tag{1}$$

and updates the voting matrix at the entries $(a_1, b_1), (a_2, b_2), (a_3, b_3)$ and (a_4, b_4) whenever $index(a_1, a_2, a_3, a_4) = index(b_1, b_2, b_3, b_4)$ Denote

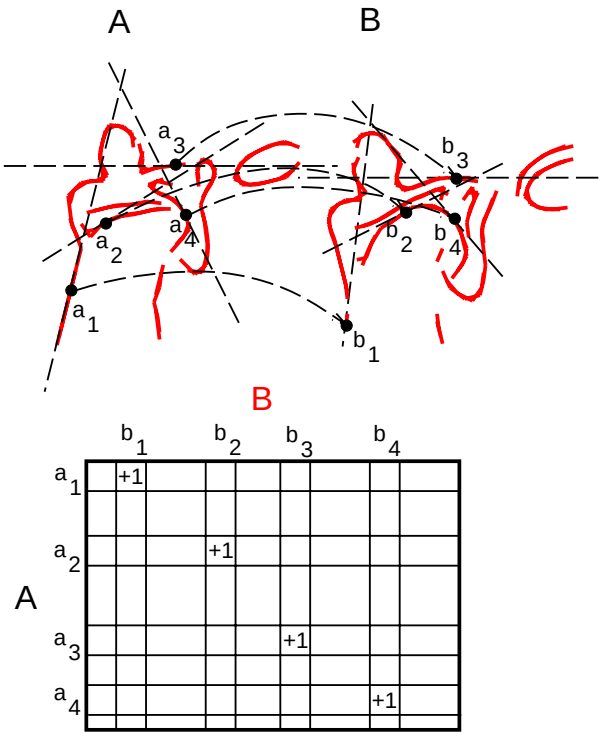

Figure 3: A voting matrix is updated whenever points $a_1\ a_2\ a_3\ a_4$ in image A and points $b_1\ b_2\ b_3\ b_4$ in image B generate point-tangent line complexes with the same topological type.

the points $i = 1, \ldots, n_a$ and the topological type indexes $k = 1, \ldots, k_{max}$. Let $A_1(i,k)$ be the number of times point i as the first point in the four point combination maps to topological type index k, i.e. the histogram of topological type indexes for point i as the first point. $A_2(i,k), A_3(i,k), A_4(i,k)$ are defined similarly, for the second, third, and fourth point respectively in the ordered four point sequence. The histogram matrices A_1, A_2, A_3, A_4 of image A and B_1, B_2, B_3, B_4 of image B contains all the information necessary to compute the voting matrix. Denote the voting matrix as V. We then get:

$$V(i,j) = \sum_{s=1}^{4} \sum_{k=1}^{k_{max}} A_s(i,k)\, B_s(j,k) \tag{2}$$

This gives a very efficient way of computing the voting matrix that avoids updating at each set of selection of four points. Only the histogram matrices have to be updated which is a comparatively simpler operation. In our applications, matching will always be relative to a key-frame image. The histogram matrices for this can therefore be precomputed.

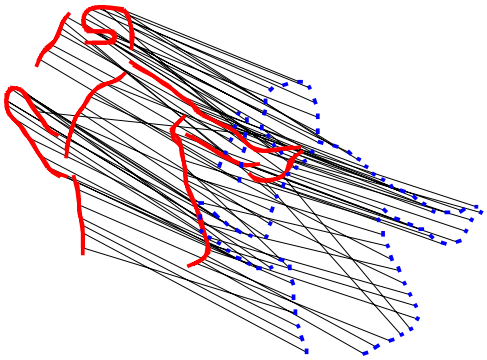

Figure 4: Matches between key-frame 251 and frame 407 in woman sequence

The histogram vectors:

$$A_l(i,k) \quad l = 1,\ldots,4 \quad k = 1,\ldots,k_{max}$$

depend on the order structure of point i relative to all surrounding triplets of points. They are determined by the context in which point i is situated. The voting matrix is obtained by inner products of these context dependent histogram vectors from images A and B. Note that this applies equally to standard geometric hashing which can be said to compare affine structure context around points. The idea of comparing shape context around specific points has recently appeared in [2] with interesting results for matching and recognition. However, the shape context descriptors in [2] have no invariance properties similar to those given by considering topological type.

4. Experimental Results

The correspondence field that is computed from the shape matching algorithm contains all the information about the relations between image A and image B. If A and B are images with a well defined deformation between them, ideally this should be captured by the correspondence field. Let $p_1^a \ldots p_n^a$ and $p_1^b \ldots p_n^b$ be the coordinates of the corresponding points in image A and B respectively and $\mathbf{T}$ the class of transformations that we know a priori defines the relation between A and B. We can base the decision that A and B are related by a member of this class of transformations on the residual:

$$\min_{T\in\mathbf{T}} \sum_{i=1}^{n} ||p_i^b - T(p_i^a)||^2 \tag{3}$$

It is important that the class of deformations is chosen to match the deformations expected. If it is chosen too large it may easily generate false positives by deforming images outside the class of interest. The class of transformations, $\mathbf{T}$, will obviously be problem dependent. In our case we want to identify a specific pose for a certain person at different time instants. The transformations should then reflect the projected image shape variation between the different instances. This transformation will obviously be quite complex, involving motions of several body parts. As a simple preliminary measure we tried various linear transformations, pure translation, similarity and affine transformations and found that pure translation gave the best results in terms of discriminating image frames. The transformation we used was therefore simply:

$$T(p_i^a) = p_i^a + t$$

and the matching distance was computed as the residual:

$$\min_t \frac{1}{n} \sum_{i=1}^{n} ||p_i^b - p_i^a - t||^2 \tag{4}$$

We believe that a more complex class of transformations modelling the projected motion of body parts, may improve the result and it will be a subject of further study

Evaluation of the complete algorithm of action recognition was made on two different sequences of tennis players called “Dennis” and “Woman” respectively. In both sequences we wanted to identify forehand and backhand strokes. The “Dennis” sequence was recorded in an actual game situation and contains both forehands and backhands intermixed, while the woman sequence was recorded at a warm up session and contains essentially only forehands. The sequences contain 3182 (2 min)and 900 (36 sec) frames respectively. For the “Dennis” sequence we selected one specific frame, 1834, see fig.

5 as the forehand key frame and 780 fig. 6 as the backhand key-frame by manual inspection. In the woman sequence we selected frame 251 fig. 7 as the forehand key frame. Note that the forehand key frames in the two sequences were chosen at different parts of the forehand action cycle. The two sequences were also shot from different angles.

The players in the sequences were tracked automatically and a window around them was cut out. On the upper half of this window we applied Canny edge detection. The edges were were tracked and subsampled to every four pixels and at each sample, the tangent direction was estimated. The number of edge points varied in general between 100- 200. No effort was made to delete the number of background edges which in some frames was substantial.By choosing just the upper half of the tracking window the decision is based on the less variable upper body posture.

The matching algorithm was applied to all frames in both sequences and a matching score for each of the three key

frames was computed (Figs.5, 6, 7). The correct matching frames are clearly visible in these figures which demonstrates the robustness of the algorithm. The figures also shows the frames representing the lowest local minima distance scores. In the "Dennis" sequence all 13 forehand strokes are found as the 13 lowest distance scores. For the backhand strokes we find 9/10 with three false positives, alternatively 7/10 with no false positives. For the "Woman" sequence all 9 forehands are found as the 9 lowest distance scores.

5. Summary and Conclusions

This paper presents an algorithm for localising action specific frames in video sequences. Its effectiveness is demonstrated by applying it to two long video sequences of tennis players. The results obtained are very promising with 100% recognition without any false positive being achieved in two of the cases examined. The crucial aspects to its success are the simplicity of the definition of an action and also the power of the matching algorithm implemented, which exploits topological type invariance.

There are, of course, obvious improvements that could be made. Firstly, a more careful choice of deformation transformation of the correspondence field could be chosen. Also a tennis stroke typically extend over 5-10 frames. Defining more than one key-frame for each action could greatly enhance robustness.

One problem not addressed is the automatic selection of the person specific key-frame. Intuitively it seems that matching person specfic key-frames should be more robust than matching a generic key-frame. These could be extracted automatically from the sequence using a person independent action specific key-frame. Preliminary results indicate this should be possible.

This method could be applied to many other sporting events for instance other racket sports, golf and basketball and would potentially be a valuable component of any sports video browsing and editing system.

References

[1] K. Akita. Image sequence analysis of real world human motion. *Pattern Recognition*, 17, 1994.

[2] S. Belongie and J. Malik. Matching with shape contexts. In *IEEE Workshop on Content-based Access of Image and Video Libraries*, June 2000.

[3] M.J. Black and Y. Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In *Proc. 5th Int. Conf. on Computer Vision*, pages 374–381, 1995.

[4] A. Bobick and J. Davis. Facial expression recognition using a dynamic model and motion energy. In *Proc. Int. Conf.on Pattern Recognition*, June 1995.

[5] A.F. Bobick and J.W. Davis. The recognition of human movement using temporal templates. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(3):257–267, March 2001.

[6] F. L. Bookstein. The measurement of biological shape and shape change. In *Lecture notes in biomathematics*. 1978.

[7] B. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 8–15, 1998.

[8] H. J. Bremermann. Cybernetic functionals and fuzzy sets. In *IEEE Systems, Man and Cybernetics Group Annual Symposium*, pages 248–254, 1971.

[9] S. Carlsson. Combinatorial geometry for shape representation and indexing. In J. Ponce and A. Zisserman, editors, *Object Representation in Computer Vision II*, pages 53–78. Springer Lecture Notes in Computer Science 1144, 1996.

[10] S. Carlsson. Order structure, correspondence and shape based categories. In *Shape Contour and Grouping in Computer Vision*, pages 58–71. Springer LNCS 1681, 1999.

[11] Y. Cui, D. Swets, and J. Weng. Learning-based hand sign recognition using shoslif-m. In *Proc. 5th Int. Conf. on Computer Vision*, June 1995.

[12] T. Darrell, P. Maes, B. Blumberg, and A. Pentland. A novel environment for situated vision and behavior. In *In Proc.IEEE Workshop for Visual Behaviors*, 1994.

[13] D.M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98, January 1999.

[14] S. Gold, A. Rangarajan, C-P. Lu, S. Pappu, and Mjolsness E. New algorithms for 2d and 3d point matching: pose estimation and correspondence. *Pattern Recognition*, 31(8), 1998.

[15] J. E. Goodman and R. Pollack. Multidimensional sorting. *SIAM J. Comput.*, 12:484–507, 1983.

[16] D. Hogg. Model-based vision: a program to see a walking person. *J. Image and Vision Computing*, 1(1):5–20, 1983.

[17] Y. Lamdan, Schwartz, and Wolfson. Object recognition by affine invariant matching. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 335–344, 1988.

[18] R. Polana and R. Nelson. Recognition of nonrigid motion. In *Proc. of ARPA Image Understanding Workshop*, pages 1219–1224, 1994.

[19] J. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. *Proc. 5th Int. Conf. on Computer Vision*, 1995.

[20] K. Rohr. Towards model-based recognition of human movements in image sequences. *Computer Vision, Graphics and Image Processing*, 59(1):94–115, 1994.

[21] S. Sclaroff. Deformable prototypes for encoding shape categories in image databases. *Pattern Recognition*, 30(4):627–640, 1996.

[22] Y. Yacoob and M. J. Black. Parameterized modeling and recognition of activities. *Computer Vision and Image Understanding*, 73(2):232–247, 1999.

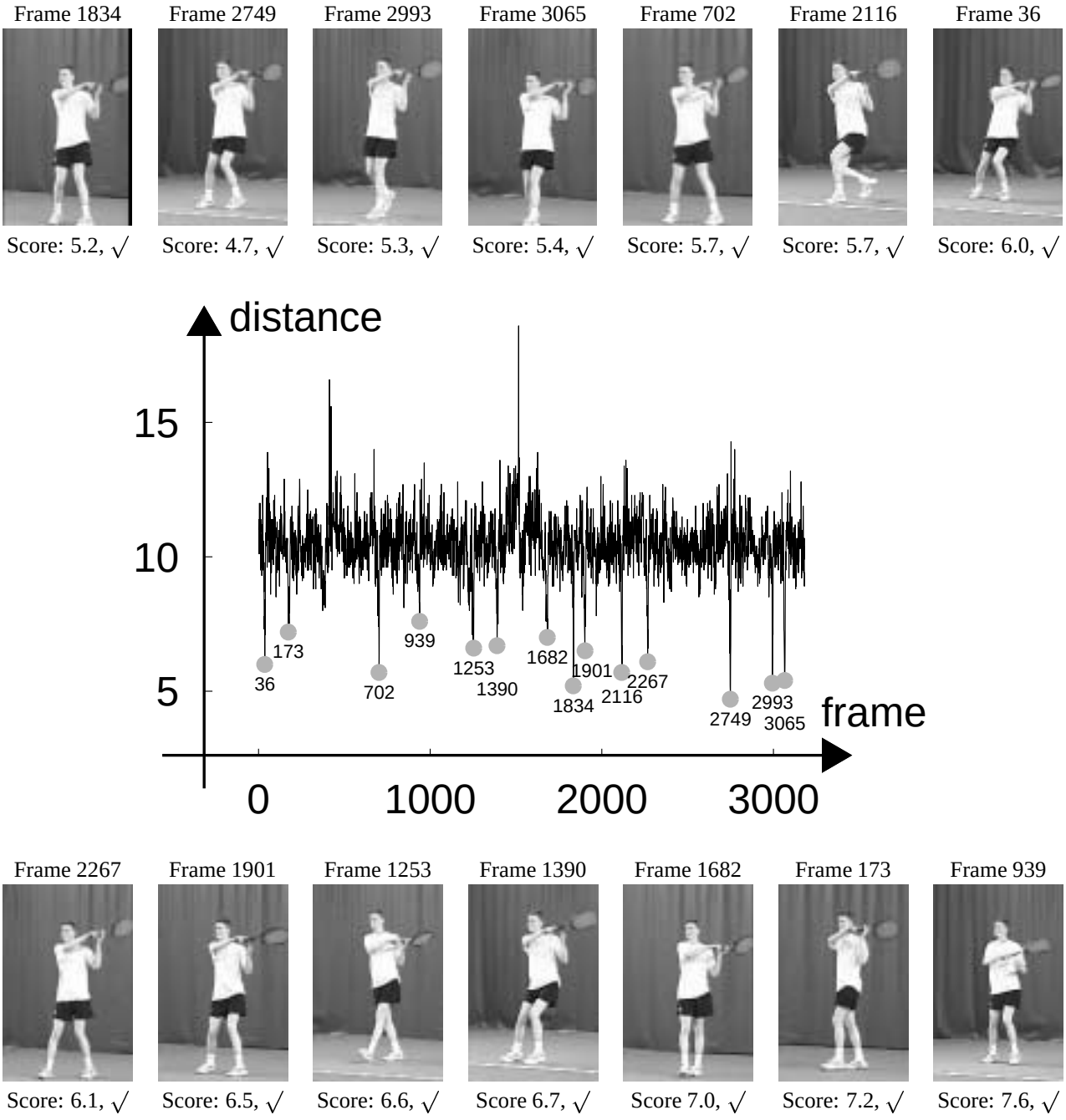

Figure 5: **Classified Forehands.** First frame displayed is used as the keyframe.

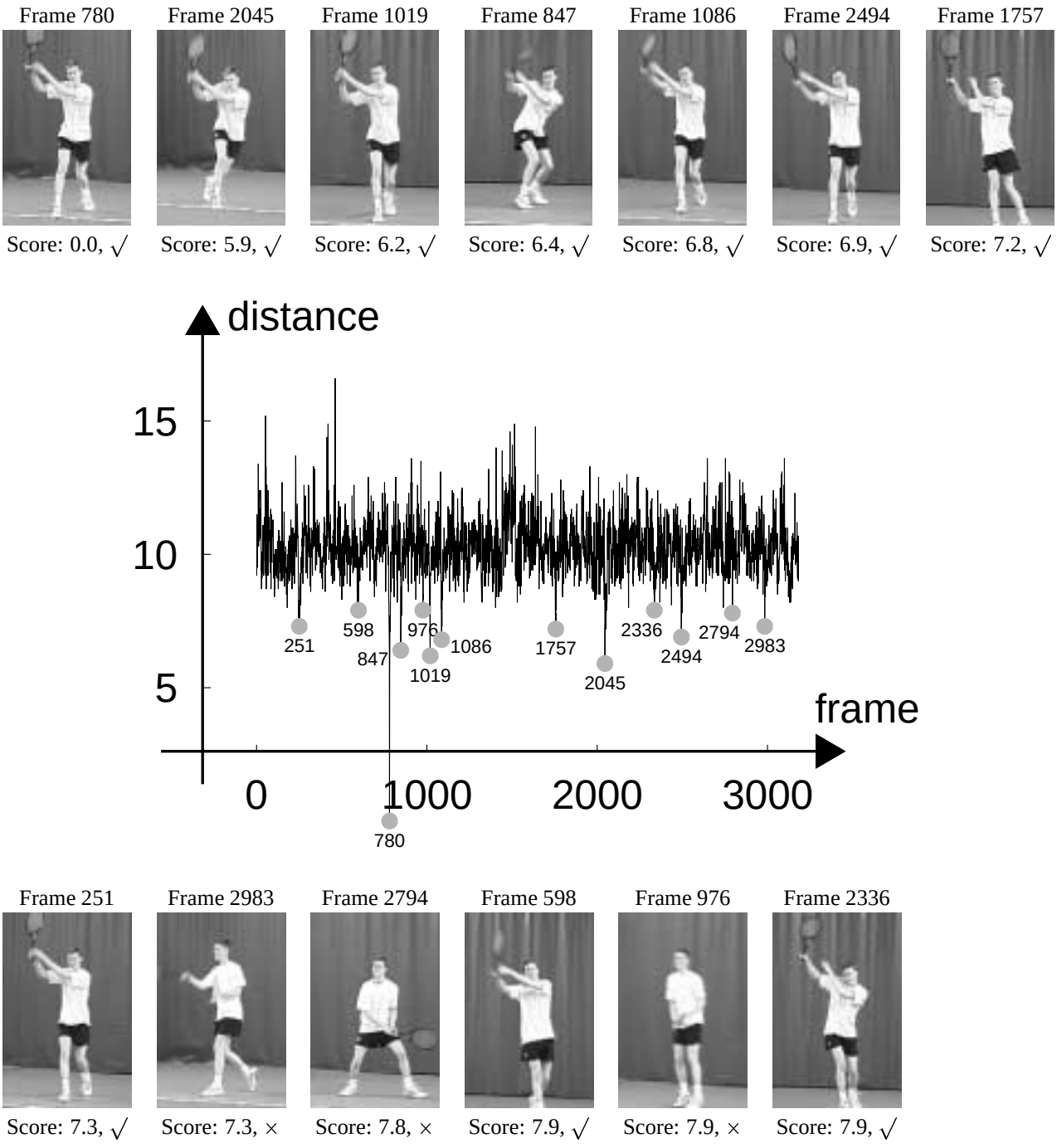

Figure 6: **Classified Backhands.** First frame displayed is used as the keyframe.

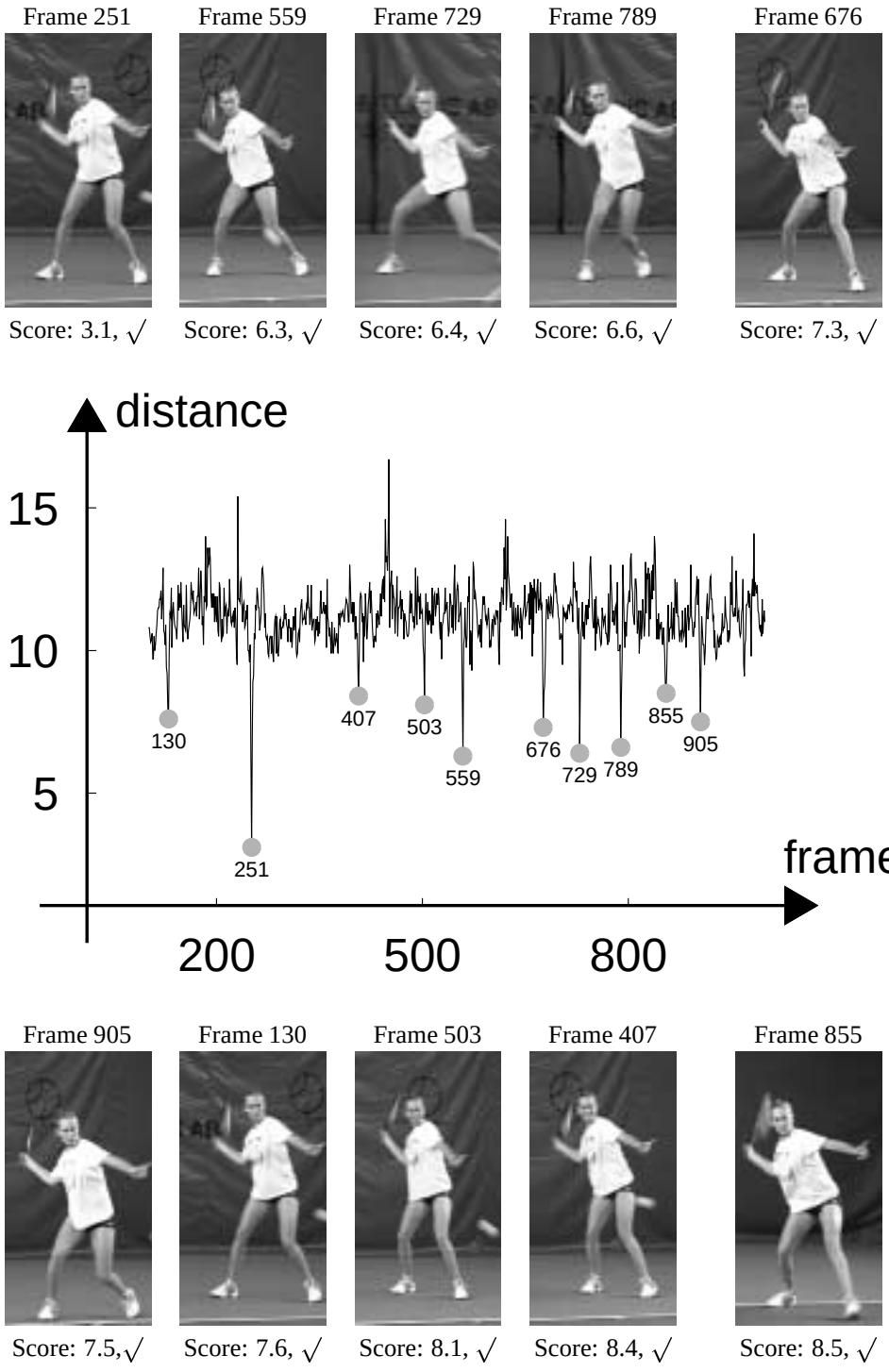

Figure 7: **Classified forehands** for the woman sequence.