

# A Rejection-Based Method for Event Detection in Video<sup>1</sup>

Margarita Osadchy and Daniel Keren

Department of Computer Science  
University of Haifa  
Haifa 31905, Israel  
E-mail: rita@research.nj.nec.com,dkeren@cs.haifa.ac.il

## Abstract

*This paper offers a natural extension of the newly introduced “anti-face” method to event detection, in both the gray level and feature domains. For the gray level domain, spatio-temporal templates are created by stacking the individual frames of the video sequence, and the detection is performed on these templates. In order to recognize the motion of features in a video sequence, the spatial locations of the features are modulated in time, thus creating a one-dimensional vector which represents the event in the detection process.*

*The following applications are presented: 1) Detection of an object under 3D rotations in a video sequence simulated from the COIL database, 2) Visual recognition of spoken words, and 3) Recognition of 2D and 3D sketched curves. The technique is capable of detecting 3D curves in viewing directions which substantially differ from those in the training set.*

*The resulting detection algorithm is very fast, and can successfully detect events even in very low resolution. Also, it is capable of discriminating the desired event from arbitrary events, and not only from those in a negative training set.*

*Possible applications of the techniques offered in this paper are in man-machine interaction, surveillance, and search and summarization in video databases.*

**Keywords:** event detection, rejectors, visual speech recognition.

Manuscript type: Transactions Letter.

Color illustrations: None.

---

<sup>1</sup> This research was supported by the Israeli Ministry of Science grant no. 1229.

## 1. Introduction

It is commonly accepted to divide the area of event detection into two parts: human action recognition and general motion-based recognition. Most of the approaches for understanding human actions require the existence of features which can be extracted from each frame of the image sequence, and then action recognition is performed on those features. Some of these techniques construct a 3D body model [23],[10],[13],[26],[28], some compute image measurements and apply temporal models to interpret the results [16],[12],[20],[3],[5].

Other related work focuses on direct motion recognition [27],[19],[6],[1],[8]. One of the interesting recently proposed venues of research is the modeling of actions by basic flow fields, estimated by principal component analysis from training sequences [1],[2]. The obvious difficulty in such an approach is that computing optical flow is non-robust, which can affect recognition results.

In this paper, both the gray level and feature domains are treated by creating spatio-temporal templates from the input sequence. We create these templates by stacking the video frames, and the detection is performed on the frame stacks. The detection algorithm is a natural extension of the newly introduced *anti-face* method [14].

Relevant research is proposed in [17] for lip reading. The sequence of images of a spoken letter was taken as a 3D template, where the third dimension is time. The authors extended the eigenface technique [24] to detect sequences of spoken letters. In this work we present detection of entire words, in lower resolution, and without restricting the negative examples to lie in a small, pre-determined set.

The lip reading task was also studied in [4],[9],[11][15],[18]. Most of the techniques extract some features of the mouth area from each frame, and then perform recognition by matching.

Another attempt to unify the spatial and temporal domains is offered in [6]. A “motion-history image” (MHI), which represents the motion at the corresponding spatial location in an image sequence, is built. This image captures only motion-related information. As mentioned by the authors, the weakness of such a technique is that in some cases it cannot discriminate between different motion directions, for instance arm-waving in opposite directions. Another drawback is that the approach will fail in the case of motion with self-occlusion. In later work [7], the recognition framework was modified by computing local motion fields from the original MHI using a gradient-based motion pyramid, and then characterizing an action by a polar histogram of motion orientations.

The advantages of our approach are:

- 1) The detection is very simple – convolution.
- 2) Negative examples are not restricted to a small, predetermined set, unlike some previous research concerning lip reading, which was restricted to collections such as some of the alphabet letters or the ten digits.
- 3) The detection performs well in very low resolution video, which is especially important for real-time applications.
- 4) The detection operates directly on the gray level frames, and does not require high-level feature extraction, which is time consuming and sensitive to noise.
- 5) The training does not require any negative examples.

Experiments have produced encouraging results. For example, the algorithm was able to discriminate the sought word (e.g. “psychology”) from similar words (e.g. “psychological”), although the training set does not contain any negative examples.

## 1.1. Structure of the Paper

Section 2 starts with a short overview of the “anti-face” technique, and then extends it to event detection by applying “anti-sequences” to frame stacks. Section 3 presents three different applications: 1) detection of an object under 3D rotations in a video sequence simulated from the COIL database, 2) lip reading, and 3) recognition of sketched curves in 2D and 3D.

## 2. Anti-Sequences

We present a fast algorithm for event detection in video sequences, which is a natural extension of anti-faces [14] to the temporal domain.

### 2.1. A Short Overview of Anti-Faces

Anti-face detectors [14] form the core of a novel detection method, which works well in the case of a rich image collection – for instance, frontal face under a large class of linear transformations, or 3D objects under different viewpoints. Call the collection of images, which should be detected, a *multi-template*. The detection problem is solved by sequentially applying very simple filters (or *detectors*), which act by inner products with a given image (viewed as a vector), and satisfy the following conditions:

- The absolute values of their inner products with the multi-template images are small.
- They are smooth, which results in the absolute values of their inner products with “random images” being large; this is the characteristic which enables the detectors to separate the multi-template from random images.
- They act independently, which implies that their false alarms are uncorrelated; hence, the false alarm rate decreases exponentially in the number of detectors.

Such detectors are found by the following process (See [14] for more details and explanation). First, choose an appropriate value  $M$  for  $\max_{t \in T} |(d_1, t)|$ , where  $d_1$  is the first detector, and  $t$  varies over a multi-template  $T$ .  $M$  should be substantially smaller than the absolute value of the inner product of two random images. Next, minimize

$$\lambda S(d_1) + \sum_{t \in T} (d_1, t)^2$$

where  $S(\cdot)$  is a roughness measure. Using binary search on  $\lambda$ , set it so that  $\max_{t \in T} |(d_1, t)| = M$ . The roughness of an  $n \times n$  image  $I$  is defined by

$$S(I) = \sum_{(k,l) \neq (0,0)}^n (k^2 + l^2) \tilde{I}^2(k, l)$$

where  $\tilde{I}(k, l)$  are the DCT (Discrete Cosine Transform) coefficients of  $I$ .

The optimization is performed in the DCT domain, and the inverse DCT transform of the optimum is the desired detector.

After  $d_1$  is found, it is straightforward to recover  $d_2$ ; the only difference is the additional condition

$$\sum_{(k,l) \neq (0,0)} \frac{\tilde{D}_1(k,l)\tilde{D}_2(k,l)}{(k^2 + l^2)^{\frac{1}{2}}} = 0$$

where  $\tilde{D}_1$  and  $\tilde{D}_2$  are the DCT transforms of  $d_1$  and  $d_2$ ; this condition guarantees that the detectors act independently. The other detectors are recovered similarly.

The detection process is very simple: an image is classified as a member of the multi-template iff the absolute value of its inner product with each detector is smaller than some (detector specific) threshold. Only images which passed the threshold test imposed by the first detector are examined by the second detector, etc. A correct choice of the threshold allows to detect not only members of the training set, but also images which are close to them.

The resulting detection algorithm is very fast; typically,  $(1+\delta)N$  operations are required to classify an  $N$ -pixel image, where  $\delta < 0.5$ .

To achieve invariance to the intensity of illumination, the images are normalized to zero mean and unit length.

## 2.2. Event Detection in Video: The Gray Level Domain

A naive approach to extending an object detection method to event detection is to perform object detection in each frame, and then classify the object's motion. Due to low resolution of the video, illumination variability, and self-occlusion, this trivial solution may be limited. In addition, in event detection we are interested more in information existing "between the frames" than in the individual frames. Hence we use an entire sequence corresponding to the sought event as a template (Figure 1). The detection process searches for the stack composed of the sequence's individual frames:

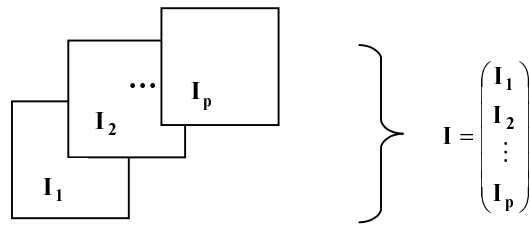


Figure 1: Stacking Frames. The sought event is represented by the sequence of video frames extending from  $I_1$  to  $I_p$ .

As explained in [14], anti-faces work because natural images are *smooth*. The same principle applies to video sequences. The basis for extending the anti-faces paradigm to event detection lies in the observation that, when viewed as a vector, the frame stack will usually be smooth. This follows from the fact that the change in natural video sequences is gradual; therefore, the function describing the variation in the temporal domain is smooth, as are the individual frames.

We create the spatio-temporal templates by stacking frames (corresponding to the event) into one vector  $I$  [17]. Next, “anti-sequence detectors” are defined as vectors satisfying the conditions listed in section 2.1. The detectors are computed as shown in section 2.1, with the definition of the roughness measure extended to the time domain:

$$S(I) = \sum_{(k,l,j) \neq (0,0,0)}^n \left( k^2 + l^2 + \left( \frac{j}{\alpha} \right)^2 \right) \tilde{I}^2(k, l, j)$$

where  $\tilde{I}(k, l, j)$  is the 3D DCT transform of  $I$ , and  $\alpha$  is a scale factor adjusting spatial and temporal “speeds”. It was chosen so that the average of the absolute values of derivative in time is equal to the average of the absolute values of derivative in the spatial domain.

Usually, a single detector is not sufficient to detect the event with no false alarms; hence we apply several detectors which act independently, as explained in [14]. The requirement that the detectors act independently implies the following condition:

$$\sum_{(k,l,j) \neq (0,0,0)} \frac{\tilde{D}_1(k, l, j) \tilde{D}_2(k, l, j)}{\left( k^2 + l^2 + \left( \frac{j}{\alpha} \right)^2 \right)^{\frac{3}{2}}} = 0$$

where  $\tilde{D}_1$  and  $\tilde{D}_2$  are the 3D DCT transforms of  $d_1$  and  $d_2$ . Once the detectors are found, the detection process proceeds in a similar manner to anti-faces.

The proposed algorithm is able to discriminate the desired event from arbitrary “natural” sequences, and the “non-events” are not restricted to a small predetermined training set of negative examples. This greatly simplifies the computation of the detectors – no database of negative examples is required – and also makes the detection more general. By “natural sequences” we mean video sequences which, on the average, vary smoothly in space and time; this covers the very large majority of sequences which are of practical importance.

There is a computational problem in the preprocessing stage, since stacking video frames results in very high dimensional vectors. Stacking one second of video with 25 fps and frame resolution of 100x60 produces a vector of dimension 150,000. One of the solutions is to compute only low frequencies of the detectors and pad the rest with zeros. However, once the detectors are recovered, their application is very fast.

### 2.3. Feature-Based Event Detection

The idea of frame stacking can also be applied to detect actions characterized by the movement of features (here, we used it to recognize symbols outlined by a laser pointer, and the feature was the pointers' image in any given frame). Each feature moving in a video sequence produces a curve  $(x(t), y(t), t)$  in the spatio-temporal domain, which we shall call an "activity curve". The activity curve contains more than the geometric structure of the curve – it is also characterized by the speed and direction in which a point moves on the curve. Extracting the spatial positions of a feature in each frame and combining them to a single vector allows to apply the anti-sequence method for detection.

First, the sequence of triplets  $(x(t), y(t), t)$  has to be converted to functions of one variable  $(t)$ . The simplest method is to define the detection of an event as the detection of both  $x(t)$  and  $y(t)$ . However, this simple approach is susceptible to symmetries in the spatio-temporal domain. For example, let us look at the case of a circle drawn counterclockwise; then,  $x(t) = \cos(t)$ ,  $y(t) = \sin(t)$ . In the case of clockwise rotation  $x(t) = \cos(t)$ ,  $y(t) = -\sin(t)$ . Since the classification is based on the absolute values of inner products between the detectors and the templates, it will not be able to discriminate between a counterclockwise and a clockwise drawn circle. To remedy this problem, we modulate  $x(t)$  and  $y(t)$  by  $t$ . For example, we can define the corresponding curves as  $x(t)+t$  and  $y(t)+t$  (this is done, of course, both in the training and detection stages).

## 3. Experimental Results

The following applications are studied: object rotation, visual speech recognition, and recognition of sketches outlined by a laser pointer. In all tests the detection was very robust; on the average, there was a difference by a factor of more than ten in the detectors' response on the positive vs. negative examples.

### 3.1. COIL Rotation Sequences

Two sets of experiments are presented. For the first test we took 20 objects (Figure 2) from the well-known COIL database (<http://www.cs.columbia.edu/CAVE/research/softlib/coil-100.html>), and simulated three types of video sequences: clockwise and counterclockwise rotation, and static. Then, for each object we built anti-sequences that discriminate clockwise rotation from other activities of the same object, or other objects. The COIL database captures the objects in five degree rotation intervals. We created a training set from sequences of length five, capturing clockwise rotation with a ten degree phase between the sequences. For example, the first sequence consists of the respective object in 0,5,10,15,20 degree angles; the second extends from 10 to 30 degrees, etc. In total, the training set included 35 sequences for each object. The test sequences for clockwise rotation were also created with a ten degree phase between the sequences, but they started with five degrees, then 15 and so on. The counterclockwise rotation and static sequences of the same object were created with 5-degree phase. Hence, the experiment included 289 sequences, all of them disjoint from the training set. Ten anti-sequences were sufficient to discriminate the clockwise rotation of each object from the counterclockwise rotation of the same object and from any activity of the other items of the COIL database, with no misclassifications. The method produced 1% of false positives in static

sequences. The reason for this is the short duration of the rotation sequences, since some of the objects hardly change during some of the sequences.

The proposed method was applied to detect rotations in a wide range of velocities, extending from 5 to 20 degrees between frames, with 4.8% of false alarms.

The next experiment was performed in order to compare the anti-sequence approach (viewed as an object detector) with anti-faces [14] applied to individual frames. The anti-face method required three to six detectors to discriminate an object from dissimilar items. To distinguish between the similar objects it usually required ten detectors, but it failed to discriminate between the objects in Figure 3.

This experiment demonstrates that anti-sequences work well not only as event detectors, but they can also enhance *object detection*, which proves that the frame stacking approach is more robust than what can be achieved by analyzing the individual frames. That is, applying anti-sequences as object detectors yields better results than applying object detection in all the individual frames.

In the second set of experiments, we addressed a more general problem: locate all instances of a particular object (a cup), performing clockwise rotation in the video sequence including the same cup performing rotations in both directions, a static cup, and several similar static and rotating objects. In this experiment the search was performed in both the spatial and temporal dimensions. Figure 4 shows fragments of the test sequence with the detection results marked by a white square around the detected image region. Six anti-sequences for the cup (Figure 5) were sufficient to correctly detect its clockwise rotation.



Figure 2: COIL subset used in rotation sequence test.



Figure 3: Anti-faces failed to discriminate between these similar objects, however anti-sequences were able to discriminate between them.

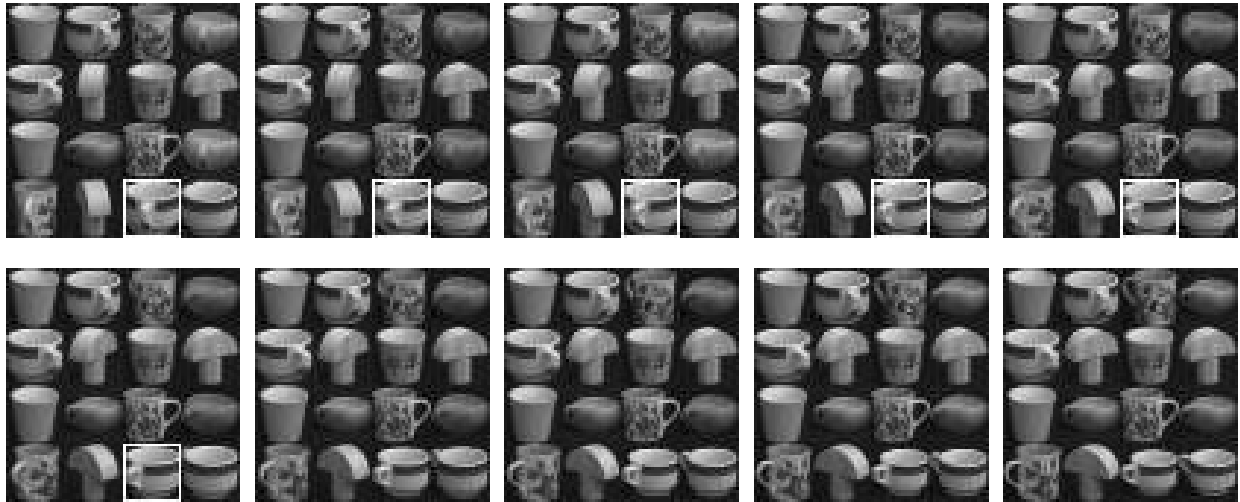


Figure 4: Fragment from the test sequence. Detection results for the 5-frame sequences of clockwise cup rotation. The white squares mark the beginning of the detected sequences.

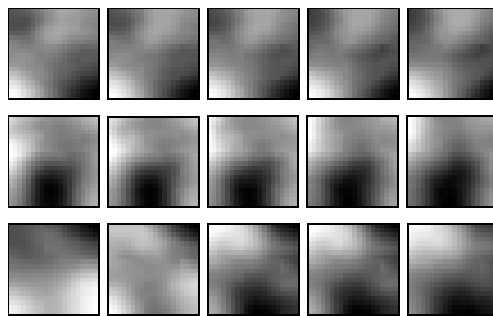


Figure 5: The first three cup anti-sequences. Note smoothness in the spatial and temporal domains.

### 3.2. Visual Speech Recognition

In this section we have tested anti-sequences in recognition of spoken words. We captured 23 sequences of the word “psychology”, uttered by a single speaker. Ten of them were used as a training set to generate the anti-sequences. The thirteen remaining sequences and twenty other words were used in the recognition test. The acquired sequences had slight variations in global head movements and changes in the duration of the articulation. To reduce these undesirable variations, spatial alignment of the mouth position and time warping of the sequences were performed in the preprocessing step. One of the “psychology” sequences was chosen as a reference (Figure 6) and all the test sequences were aligned and warped against it. The resulting sequences contained 26 images downsized and cropped to 24x16 pixels and centered around the lips. Since the mouth is symmetric in the  $x$ -direction, we used only half of the images.

#### 3.2.1. Spatial Alignment

Individual frames were aligned against the first frame of the reference sequence by extracting distinct features on the face, and warping the frames by a corresponding rigid transformation.

### 3.2.2. Temporal Warping

Temporal warping was performed in the same way as described in [17]. The algorithm is based on the dynamic programming algorithm of Sakoe and Chiba [21].

Let  $W$  be the reference sequence with size  $N$ , and let  $A$  be an input sequence with size  $M$  that should be warped to size  $N$ . The warping algorithm uses the *DP-equation* in symmetric form with a slope constraint of 1:

$$g(i, j) = \min \begin{bmatrix} g(i-1, j-2) + 2d(i, j-1) + d(i, j) \\ g(i-1, j-1) + 2d(i, j) \\ g(i-2, j-1) + 2d(i-1, j) + d(i, j) \end{bmatrix}$$

where  $d(i, j) = \|W_i - A_j\|$  is the distance from the  $i$ -th element of the sequence  $W$  to the  $j$ -th element of the sequence  $A$ . The initial conditions are:

$$\begin{aligned} g(1,1) &= 2d(1,1) \\ d(i, j) &= \infty, \quad g(i, j) = \infty \quad \text{for } i, j \leq 0 \end{aligned}$$

The minimal argument chosen for the calculation of  $g$  at the point  $(i,j)$  defines the path from the previous point to the current one, thus creating a path from  $(1,1)$  to  $(M,N)$ . Each point on the path indicates which frames from the input sequence match to frames in the reference sequence. In the case of two frames from the input sequence matching to one frame in the reference sequence, they are averaged to create a single frame. If one frame from the input sequence matches two frames from the reference sequence, it is duplicated. At the end of this process, the input sequences are warped to the size of the reference sequence.

### 3.2.3. Results

The experiment's goal was to recognize the word "psychology" in a test set that contained thirteen instances of "psychology" which did not appear in the training set, and twenty other words. The words tested for recognition were: "crocodile", "dinosaur", "encyclopedia", "transform", "integrable", "associative", "homomorphism", "leadership", "differential", "deodorant", "commutative", "anthropology", "trigonometry", "psychological", "anthology", "astrology", "cardiology", "dermatology", "genealogy", "university". We have chosen the words such that some of them are totally different from "psychology" (like "crocodile" in Figure 7), one word is very similar – "psychological" (Figure 8), and the others have the same suffix ("ology") like the sought word (Figure 9 shows the word "anthology").

Three anti-sequences (Figure 10) sufficed to recognize all instances of "psychology" in the test set, with no misclassifications.



Figure 6: Reference sequence for the word "psychology".

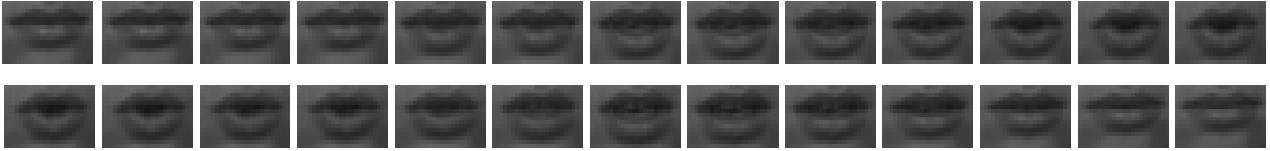


Figure 7: The word "crocodile" warped to the length of the reference sequence.

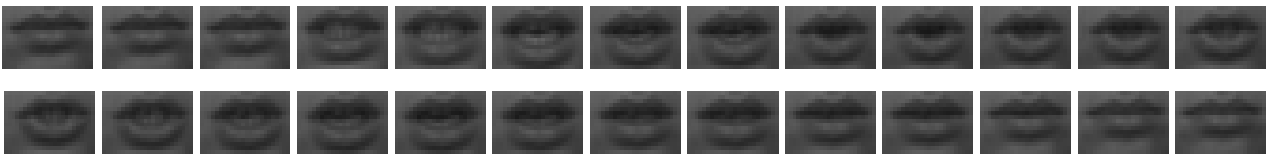


Figure 8: The word "psychological" warped to the length of the reference sequence.

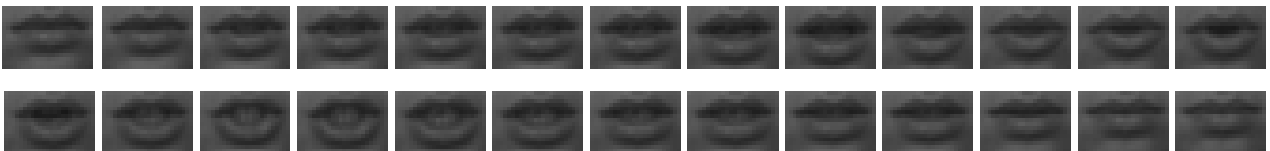


Figure 9: The word "anthology" warped to the length of the reference sequence.

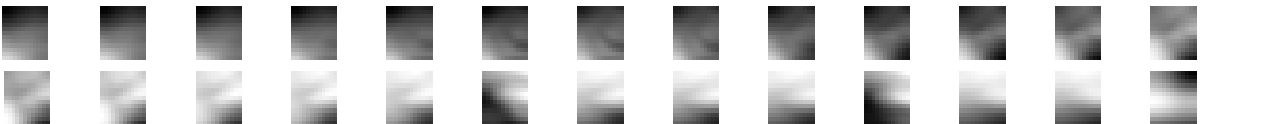


Figure 10: First anti-sequence (of three) for the word "psychology". Since the mouth is symmetric, the anti-sequence frames are half the size of the images.

### 3.3. Symbol/Sketch Detection

This experiment concerned recognition of sketched symbols, and the results can also be applied to gesture detection etc. Various symbols were outlined with a laser pointer on a white background, and the process was captured on video. Then, the symbols were segmented from the background, and linear interpolation was applied in order to normalize them in space and time, which yielded a representation of each symbol as a vector with 200 points. Anti-sequence detectors were then constructed and applied to the one-dimensional vectors. The sought symbol was the infinity sign drawn at a certain order, shown in Figure 11(b). The test set contained this symbol as well as the infinity sign drawn in a different direction, and other symbols (Figure 11):  $\alpha$ ,  $\beta$ ,  $\gamma$ , circle, square, and the digits 6, 8, 9. Two detectors sufficed to correctly detect the sought sketch with no false alarms, in

all 50 tests performed. As Shown in Figure 11 (a), detection was robust under both local and global deformations in the curve, which result from it being outlined by hand. Such deformations pose considerable difficulty for recognition methods which use differential invariants, as they are very susceptible to local distortions.

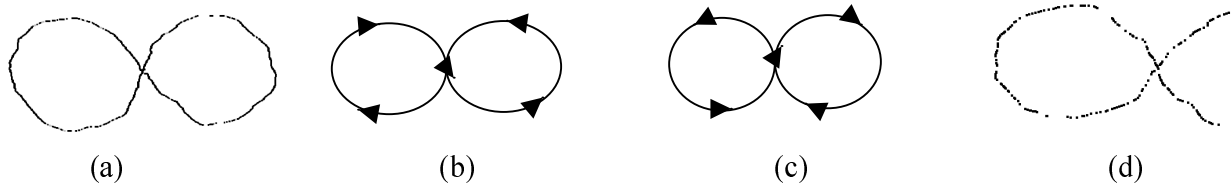


Figure 11: (a) one of the “infinity” sequences used for training; (b) and (c) schematic drawings of the infinity symbol traversed in two different directions; (d) the symbol  $\alpha$ , one of the negative test examples.

### 3.4. Sketch Detection in 3D

So far, we have discussed detection in which the training set contained sequences which approximate the sought sequence. In some cases, this assumption does not hold. Consider, for example, the problem of recognizing a sketched symbol which is three-dimensional (i.e. unlike the infinity symbol in Section 3.3, the sketch is not confined to a plane). This problem may arise, for example, in gesture detection; a certain feature, such as the tip of a finger, may outline a non-planar curve. Suppose also that we do not know the angle from which the sketch was photographed. In principle, this problem could be alleviated by preparing a very large training set, which includes a dense sampling of the viewing sphere. However, this is very time consuming. The method presented here allows to detect different views using only a very small number of samples, by using the fact that a small number of views (*basis views*) span the entire view space. If a detector has a small inner product with these basis views, it will also have a small inner product with sequences that are spanned by them. Hence these sequences will also be detected.

The basic result we rely on appears in [25], where it is proved that for a transparent object in 3D, the  $x$  and  $y$  coordinates of all views can be expressed as a linear combination of two basis views. If a detector is suitable for these views, it will also detect other views (unless they are taken at angles in which the projection of the object is highly singular). In order to apply the method, corresponding points have to be chosen between the candidate view and the basis views. This was achieved by normalizing the curves to the same length, and using 200 evenly distributed points in both curves as the pairs of corresponding points. Since Euclidean length is not preserved between two projections of the same curve, we have used a measure of length which is invariant to affine transformations [22]. This was sufficient for a rather wide range of viewing angles.

We have tested these assumptions on the 3D curve which is depicted in Figure 12. The curve was sketched with a laser pointer on a curved piece of cardboard. The process was filmed by a video camera, and the curve's points extracted from the individual frames (note that, as before, the curve has an order and rate of traversing associated with it). The two basis views are depicted in Figure 13. The process was repeated with the camera positioned at other angles, and some of the resulting sketches which were detected are depicted in Figure 14. In Figure 15 some curves from the negative test set are shown.

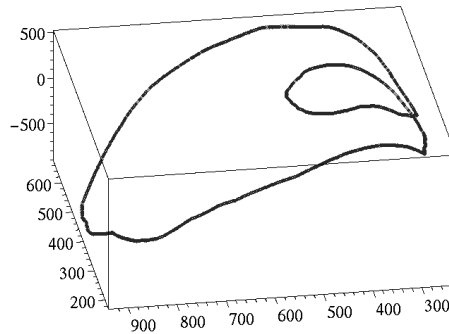


Figure 12: A curve used for testing the detection scheme.

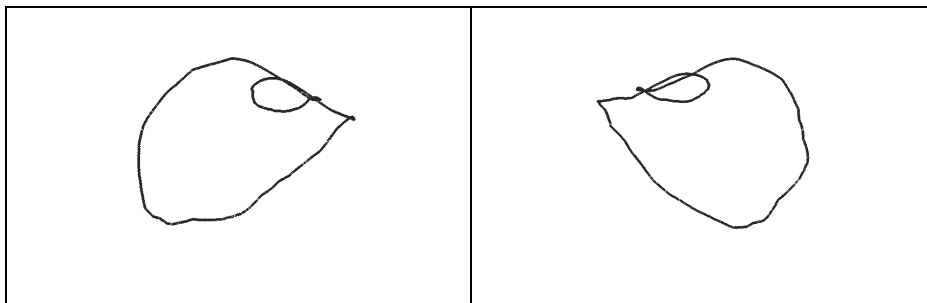


Figure 13: Two basis views for the curve in Figure 12.

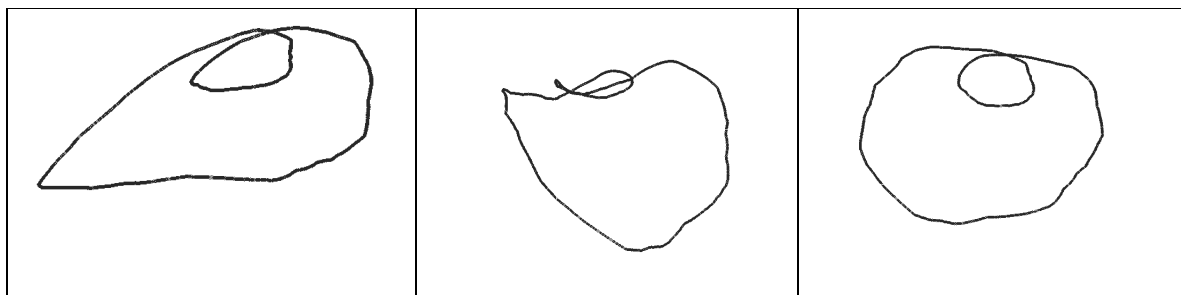


Figure 14: Three of the curve views that were detected by the algorithm

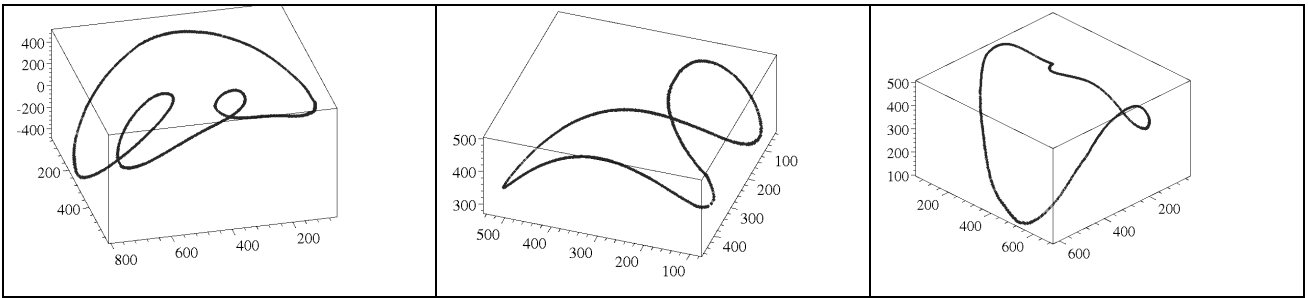


Figure 15: Three curves from the negative test set.

Three detectors were required to successfully detect all the views of the curve in Figure 12 that were tested, against ten other curves (negative examples), three of which are depicted in Figure 15. To demonstrate the nature of the detection process, we have included in Figure 16 the results of applying the first two detectors to ten views of the sought curve, compared with the results for ten views of one of the negative examples (center curve in Figure 15). Note that there is one false alarm for each detector (views 6, left and 5, right), for which the corresponding detector yielded a small result, but no negative example passed the combined test of the two detectors. The threshold (scaled to the proportions in Figure 16) was 8.0.

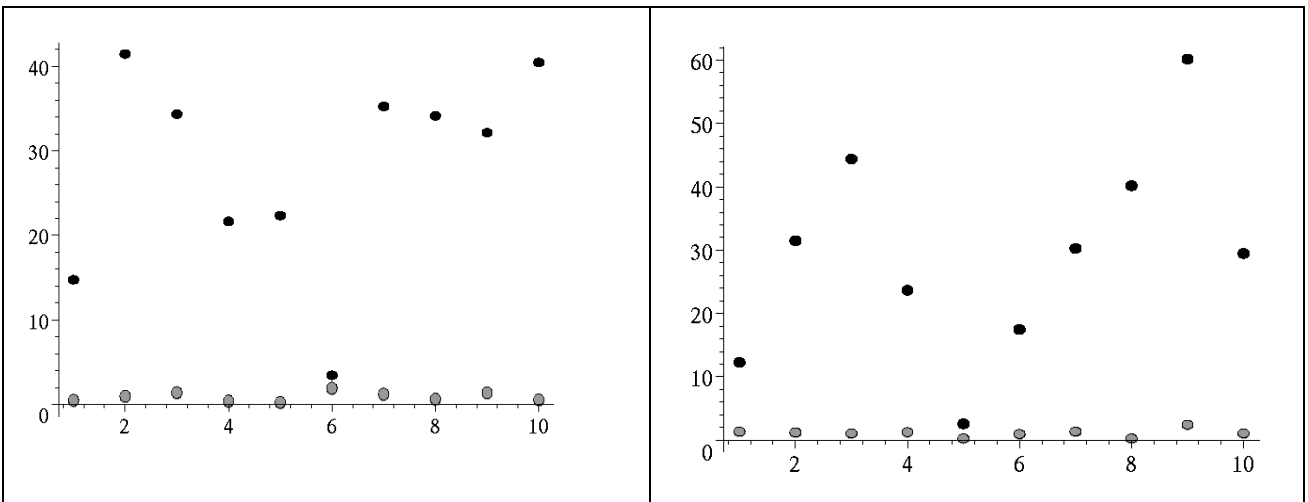


Figure 16: Results of applying the first (left) and second (right) detectors to ten views of the sought activity curve (results depicted in gray) and the negative example in Figure 15, center (results depicted in black). Horizontal axis stands for view number, vertical axis for detector's output, scaled by 1000. While each detector admits one false alarm, the intersection of the sets of curves admitted by each of them successfully separates all the views of the sought curve from all the negative example's views.

## 4. Conclusions

The “anti-face” method was extended to the time domain, and used to detect events in video sequences. The algorithm was tested on sequences of rotating objects, and it was demonstrated that detection is more successful than separate detection in each frame, and that it can detect rotations over a wide range of velocities. Another example was the detection of spoken words in a video sequence; the algorithm performed well, although the resolution was very low. The set of negative examples was not restricted, and contained words similar to the sought word. The algorithm was also applied to detect “activity curves”, which correspond to sketched symbols in 2D and 3D. Using two “basis views”, it was possible to successfully detect sketches in views that substantially differ from the training set views.

Future research will concentrate on expanding the method to detect “generic” activity (e.g. walking people).

## 5. References

- [1] M.J. Black, D.J. Fleet, Y. Yacoob. “Robustly estimating changes in image appearance”, *Computer Vision and Image Understanding* vol. 78, pp.8-31, 2000.
- [2] M.J. Black, Y. Yacoob, A.D. Jepson and D.J. Fleet. “Learning parameterized models of image motion,” *IEEE Conf. on Computer Vision and Pattern Recognition*, 1997.
- [3] C. Bregler. “Learning and recognizing human dynamics in video sequences,” *IEEE Conf. on Computer Vision and Pattern Recognition*, 1997.
- [4] C. Bregler and Y. Konig. “Eigenlips for robust speech recognition,” *Proc. IEEE Inter. Conf. on Acoustics, Speech, and Signal Processing*, 1994.
- [5] S. Carlsson. “Recognizing walking people,” *European Conf. on Computer Vision*, pp. 472-486, 2000.
- [6] A.F. Bobick and J.W. Davis. “The recognition of human movement using temporal templates,” *IEEE Trans. on Pattern Anal. and Mach. Intell.*, vol. 23 (3), 2001.
- [7] J. W. Davis.”Hierarchical motion history images for recognition of human motion,” *Proc. IEEE Workshop on Detection and Recognition of Events in Video*, pp. 39 – 46, July 2001.
- [8] I.A. Essa and A. P. Pentland. “Facial expression recognition using a dynamic model and motion energy,” *Proc. Intern. Conf. on Computer Vision*, 1995.
- [9] K.E. Finn and A.A. Montgomery. “ Automatic optically-based recognition of speech,” *Pattern Recognition Letters*, 8:159-164,1988.
- [10] D.M. Gavrila and L.S. Davis. “3-D model-based tracking of humans in action: A multi-view approach”, *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 73-80, 1996.
- [11] A.G. Goldschen. “*Continuous automatic speech recognition by lipreading*,” PhD thesis, George Washington University, School of Engineering and Applied Science, 1993.

- [12] D. Hogg. "Model-based vision: A program to see a walking person," *Image and Vision Computing*, 1(1):5-20, 1983.
- [13] I. Kakadiaris and D. Metaxas. "Model-based estimation of 3D human motion with occlusion based on active multi-viewpoint selection," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 81-87, 1996.
- [14] D. Keren, M. Osadchy, and C. Gotsman. "Anti-Faces: A novel, fast method for image detection," *IEEE Trans. on Patt. Anal. And Mach. Intelligence*, 23(7): 747 – 761.
- [15] M. Kirby, F. Weisser, and G. Dangelmayr. "A model problem in the representaiton of digital image sequences," *Pattern Recognition*, 26(1):63-73, 1993.
- [16] M.E. Leventon and W.T. Freeman. "Bayesian estimation of 3-d human motion from image sequence", TR-98-06, Mitsubishi Electric Research Lab, 1998.
- [17] N. Li, S.Dettmer, and M. Shah. "Visually recognizing speech using eigensequences", *Motion-Based Recognition, Kluwer Academic Publishing*, pp. 345-371,1997.
- [18] K. Mase and A. Pentland. "Lip reading: Automatic visual recognition of spoken words", TR 117, M.I.T. Media Lab Vision Science, 1989.
- [19] R. Polana and R. Nelson. "Low level recognition of human motion," *IEEE Workshop on Nonrigid and Articulated Motion*, 1994.
- [20] J.M Regh and T. Kanade. "Model-based tracking of self-occluding articulated objects," *Proc. Intern. Conf. on Computer Vision*, 1995.
- [21] H. Sakoe and S.Chiba. "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. on Acoustic, Speech, and Signal Processing*, 26(1):43-49, February 1978.
- [22] E. Salkowski, *Affine Differential Geometry*. de Gruyter & Co., Berlin 1934.
- [23] H. Sidenbladh, M.J. Black, and D.J. Fleet. "Stochastic tracking of 3D human figures using 2D image motion," *European Conf. on Computer Vision*, pp. 702-718, 2000.
- [24] M. Turk and A. Pentland, "Eigenfaces for Recognition," *Journal of Cognitive Neuroscience*, 3 (1):71--86, 1991.
- [25] S. Ullman and R. Basri, "Recognition by linear combinations of models", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10): 992-1006, 1991.
- [26] S. Watcher and H.H. Nagel. "Tracking persons in monocular image sequences," *Computer Vision and Image Understanding*, " vol. 74, no.3, pp. 174-192, 1999.
- [27] Y. Yacoob and M.J. Black "Parameterized modeling and recognition of activities," *Computer Vision and Image Understanding*, vol 73, no. 2, pp 232-247, 1999.
- [28] M. Yamamoto, A. Sato, S. Kawada, T. Kondo, and Y. Osaki. "Incremental tracking of human actions from multiple views," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2-7, 1998.