

Robust Online Change-point Detection in Video Sequences

Gabriel Tsechpenakis, Dimitris N. Metaxas

CS Department,
Rutgers University,
Piscataway, NJ 08854
gabriel.t, dnm@cs.rutgers.edu

Carol Neidle

Dept. of Modern Foreign
Languages and Literatures,
Boston University,
Boston, MA 02215
carol@bu.edu

Olympia Hadjiliadis
Electrical Engineering Dept.,
Princeton University,
Princeton, NJ 08544
ohadjili@Princeton.edu

Abstract

We present the Cumulative Sum (CUSUM) stopping rule, applied to Computer Vision problems, to automatically detect changes in either parametric or nonparametric distributions, online or off-line. Our approach is based on using the previously received data of the sequence to detect a change in data that are to be received. We assume that no significant change has occurred up to an unknown time instance. Then a change in the distribution of the observations occurs and the objective is to estimate this instance. We test the hypotheses of no change occurs vs. a change occurs at the current frame, which is done by the CUSUM stopping rule. We apply our framework to the case of continuous 3D hand tracking, where the high dofs, the fast finger articulations, the large rotations and the frequent occlusions often cause error accumulation. Also we illustrate the performance of our approach in video segmentation, and specifically in segmentation of fingerspelling in American Sign Language (ASL) videos.

1. Introduction

The detection of abrupt changes is a crucial open problem. So far, methods using parametric distributions, such as Kalman Filter [15], and nonparametric distributions, such as the CONDENSATION algorithm [6], make assumptions on temporal continuity, e.g. motion smoothness, and cannot formally deal with abrupt changes. Therefore, the detection of such changes and coupling or switching between different methods is necessary.

We present an approach to detecting changes in videos and coupling between different methods. The main motivations of our work are: (i) to detect abrupt changes in sequences online, (ii) to detect change-points in noisy signals, (iii) to obtain reliable temporal boundaries when we do not have training samples, and (iv) in cases where training data are available, to assist other methods, such as HMMs [12], when abrupt changes occur and the continuity assumptions do not hold [3].

We introduce to Computer Vision the Cumulative Sum (CUSUM) stopping rule [4], a statistical procedure that detects changes in both parametric and nonparametric distributions, either online or off-line. CUSUM assumes that the input signal is *in-control*, i.e. no significant change has occurred up to an unknown time instance. Then a change in the input signal occurs and the objective is to estimate this instance. For each frame, we test the hypotheses of *no change occurs* vs. *a change occurs at the current frame*, which is done by the CUSUM stopping rule.

We focus on two major tasks of Computer Vision, namely object tracking and video segmentation. For the case of object tracking, we focus on the 3D hand tracking, where the high *dofs*, the fast finger articulations, the large rotations and the frequent occlusions cause error accumulation, which is difficult to overcome; therefore, the need for automatic tracker re-initialization emerges and this can be done by using a discrete tracking method. For the case of video segmentation, we focus on American Sign Language (ASL) videos. In this case, we use the hand shape changes to discriminate between different linguistic phases, namely the fingerspelling and the continuous signing [19].

We chose to use this application for our approach due to the large segmentation ground-truth information available. Furthermore, fingerspelling segmentation is a crucial task for the ASL (and all sign languages) recognition [20]. This is because the internal linguistic structure of these two phases differs significantly, and thus the strategies required for recognition of these signs must differ accordingly. This is an example where HMMs, which have been used in the past for ASL recognition [20], and our approach can be coupled into an integrated recognition scheme.

In Section 2 we describe the previous work on change-point detection, 3D hand tracking and ASL recognition. Due to the limited space, we could not include general literature on tracking and video segmentation. In Section 3 we describe the CUSUM framework, in 4 we describe the two indicative applications, namely the hand tracking (4.1) and the ASL video segmentation (4.2). In the same Section, we show the results of our approach on the specific input signals. Finally, in Section 5 we give our conclusions.

2. Previous Work

One of the major problems in Computer Vision is 3D tracking of articulated objects, such as hands [13, 7, 18, 1]. Hand tracking is an indicative example of the difficulties emerging, namely large rotations, fast movements (finger articulations) and occlusions, mainly due to the hands' high *dofs*.

There are generally two major approaches for tracking: (i) parametric, such as Kalman Filter [15], where a single parametric distribution models the input observations, and (ii) nonparametric, where the input observations are modeled based on nonparametric [6] or mixtures of parametric distributions. Among these approaches, there are (i) continuous methods [9, 22, 7, 13, 18, 17] that use both temporal and static information from the input sequence, and (ii) discrete methods [1, 16], which handle each frame separately, using only static information and some kind of prior knowledge. Continuous trackers provide high accuracy and low complexity, exploiting the continuity constraints over time. When the assumptions of smooth motion do not hold, continuous trackers may fail, and they usually cannot recover easily. On the other hand, discrete approaches do not suffer from error accumulation over time, giving independent solutions at each time instance, but their accuracy depends on the generality of the prior knowledge they utilize and they are usually time consuming. For robust continuous tracking over long time, automatic tracker re-initialization or coupling between continuous and discrete tracking is necessary. Therefore the *change-point*, i.e. when the continuous tracker starts losing track, needs to be detected.

Another problem in Computer Vision is the video segmentation. An example of video segmentation is found in the recognition of Sign Language. So far, HMMs have been

used only for continuous signing recognition [20], whereas recognition strategies for fingerspelling [5] must be fundamentally different. However, fingerspelling segmentation within a fluent stream of signing is non-trivial, as many of the same handshapes that are used as letters are also used in the formation of other types of signs. Therefore the *change-points*, i.e. when a fingerspelling phase begins and ends, need to be detected.

Page [11] proposed the CUSUM test, in connection with industrial quality control, for change-point detection. The problem of optimal sequential change-point detection was solved by Shiryaev [14], who proposed solution both for discrete and continuous time in a Bayesian framework. The CUSUM test was proven optimal, in the minmax Lorden sense [8], by Moustakides in 1986 [10]. A good example of the generality of the CUSUM procedure is the work of Chen et. al. [3], where the change-point detection assists the coupling (switching) between different HMMs.

3. Our Method

We apply the CUSUM procedure for the detection of temporal changes in video sequences. We describe how this framework is applied to (i) tracking, and specifically to the case of the 3D hand tracking, and (ii) video segmentation, focusing on the case of fingerspelling segmentation in ASL videos.

The main advantages of our framework are: (i) it detects change-points online, (ii) it exploits statistical measurements of the input signal without the need of training samples, (iii) it is robust to noise, (iv) it can assist existing dynamic learning methods when abrupt changes violate continuity assumptions, and (iv) it can be used in a variety of Computer Vision tasks, where changes in sequential data need to be detected on- or off-line.

Consider the case of object tracking, where we have X_1, \dots, X_n observations from the input frames of a video. Suppose our tracker is initially *in-control*, i.e., the variation of these observations is due to an assignable cause, and hence the tracker can handle it. At an unknown instance τ , the tracker goes *out-of-control*, i.e., starts failing. The objective is to detect the change from *in-control* to *out-of-control* as soon as possible.

3.1.1 Mathematical formulation

Assume that the tracker fails at an unknown time $\tau > 1$. Also assume that the input observations $X_1, \dots, X_{\tau-1}$ are independent random variables with a probability density function $f_0(X_i)$, while the observations $X_\tau, X_{\tau+1}, \dots$ are independent random variables with a probability density

function $f_1(X_i)$. The probabilistic setting of the problem can be summarized as follows: (i) P_τ is the probability that the change from f_0 to f_1 occurs at time τ , and (ii) P_0 is the probability that the change from f_0 to f_1 never occurs. Thus, our problem consists of testing the composite hypotheses:

$$\left\{ \begin{array}{l} H_0 : \text{The tracker never fails} \\ \text{vs.} \\ H_1 : \text{The tracker fails at time } \tau = 1, \\ \text{or} \\ H_2 : \text{The tracker fails at time } \tau = 2, \\ \text{etc.} \end{array} \right. \quad (1)$$

3.1.2 The CUSUM stopping rule

Suppose that we have obtained X_1, \dots, X_n observations up to an instance n , and we test the above hypotheses for these observations. According to the Neyman-Pearson lemma for conducting any simple hypothesis test [2], the *Uniformly Most Powerful test* is the one for which we reject the null hypothesis (H_0) whenever,

$$\sum_{i=\tau}^n \log \frac{f_1(X_i)}{f_0(X_i)} > \nu, \quad (2)$$

assuming the observations are independent variables. This suggests that for the testing of eq.(1), a reasonable test would be to reject the null hypothesis whenever

$$\left\{ \begin{array}{l} \text{either } \sum_{i=1}^n \log \frac{f_1(X_i)}{f_0(X_i)} > \nu, \\ \text{or } \sum_{i=2}^n \log \frac{f_1(X_i)}{f_0(X_i)} > \nu, \\ \dots \\ \text{or } \sum_{i=n}^n \log \frac{f_1(X_i)}{f_0(X_i)} > \nu \end{array} \right. \quad (3)$$

The above is equivalent to rejecting the null hypothesis whenever

$$\max_{0 \leq \tau \leq n} \sum_{i=\tau}^n \log \frac{f_1(X_i)}{f_0(X_i)} > \nu \quad (4)$$

assuming the log likelihood ratio $\log \frac{f_1(X_0)}{f_0(X_0)} = 0$. This gives rise to the following stopping rule,

$$T_c = \min\{n : \max_{0 \leq \tau \leq n} \sum_{i=\tau}^n \log \frac{f_1(X_i)}{f_0(X_i)} > \nu\}, \quad (5)$$

where T_c is the time when the tracker fails.

Definition 1 Let $S_j = \sum_{i=1}^j \log \frac{f_1(X_i)}{f_0(X_i)}$ and $S_0 = 0$. Then we have:

1. The CUSUM statistic process:

$$S_n = \min_{0 \leq k \leq n} S_k \quad (6)$$

2. The CUSUM stopping rule:

$$T_c = \min\{n : S_n - \min_{0 \leq k \leq n} S_k > \nu\} \quad (7)$$

Moreover, let us introduce a computation-friendly version of the CUSUM stopping rule. Let $g(X_i) = \log \frac{f_1(X_i)}{f_0(X_i)}$ and define $D_n = \max\{0, D_{n-1} + g(X_n)\}$ with $D_0 = 0$. Then the stopping rule of eq. (7) becomes,

$$T_c = \min\{n : D_n > \nu\} \quad (8)$$

3.1.3 Optimality of the CUSUM stopping rule

The objective is to detect a change from successful tracking (*in-control*) to loss of track (*out-of-control*), as soon as possible, while controlling the frequency of false alarms γ . Thus, our aim is to minimize over all stopping rules the worst detection delay [8],

$$J(T) = \sup_{\tau} E_{\tau}[(T - (\tau - 1))^+ | \sigma\{X_1, \dots, X_{\tau-1}\}], \quad (9)$$

subject to the frequency of false alarm constraint ($\gamma > 0$), $E_0[T] \geq \gamma$, where $E_0[T]$ denotes the expectation of having false alarm while *in-control* (distribution f_0).

In the above formulation, $(T - (\tau - 1))^+$ is the detection delay of our stopping rule and it is $(T - (\tau - 1))^+ = \max\{0, (T - (\tau - 1))\}$. Also, $\sigma\{X_1, \dots, X_{\tau-1}\}$ is the sigma algebra generated by the observations, i.e., it is the information carried after having observed everything up to time $\tau - 1$. In this expression the detection delay $(T - (\tau - 1))^+$ is projected based on what is observed up to time $\tau - 1$. Obviously, the conditional expectation $E_{\tau}[(T - (\tau - 1))^+ | \sigma\{X_1, \dots, X_{\tau-1}\}]$ is a random variable, since it depends on the observations $X_1, \dots, X_{\tau-1}$. Thus, the above expression gives us the essentially largest value that one can get for the expected delay given the different $X_1, \dots, X_{\tau-1}$ one can have as observations.

Notice that, due to the fact that the CUSUM statistic is always non-negative (see definition 1), the worst detection delay over all possible observations up to time τ , and all possible change points τ , will occur whenever, at time τ , the CUSUM statistic takes the value 0.

As proven by Moustakides in 1986 [10], the CUSUM stopping rule is the optimal solution to the above problem, where the threshold ν is chosen so that $E_0[T] = \gamma$.

3.1.4 Parametric Considerations

Consider the case that $f_0(X_i)$ and $f_1(X_i)$ belong to the one-parameter exponential family of distributions. In other

words, let $f_0(X_i) = h(X_i) \cdot \exp\{\theta_0 t(X_i) - \psi(\theta_0)\}$ and $f_1(X_i) = h(X_i) \cdot \exp\{\theta_1 t(X_i) - \psi(\theta_1)\}$, where $t(X_i)$ is the sufficient statistic. Then, the CUSUM stopping rule is calculated from eq. (7) (for $l = n, k$),

$$S_l = (\theta_1 - \theta_0) \sum_{i=0}^l t(X_i) - l \cdot (\psi(\theta_1) - \psi(\theta_0)), \quad (10)$$

In particular, if f_i is a gaussian $N(\theta_i, \sigma^2)$, for $i = 0, 1$, where σ^2 is assumed to be known, then the sufficient statistic $t(X_i) = X_i$ and $\psi(\theta_i) = \theta_i^2$, and the CUSUM stopping rule is calculated from eq. (7) ($l = n, k$),

$$S_l = 2(\theta_1 - \theta_0) \sum_{i=0}^l X_i - l \cdot (\theta_1^2 - \theta_0^2), \quad (11)$$

In order to computationally simplify eq. (7) under these assumptions, we distinguish the following two cases:

1. $\theta_1 > \theta_0$: Let $d_n = \max\{0, d_{n-1} + (X_n - \frac{\theta_1 + \theta_0}{2})\}$ and $d_0 = 0$, then

$$Tc = \min\{n : d_n > \nu\} \quad (12)$$

2. $\theta_1 < \theta_0$: Let $e_n = \max\{0, e_{n-1} - (X_n - \frac{\theta_1 + \theta_0}{2})\}$ and $e_0 = 0$, then

$$Tc = \min\{n : e_n > \nu\} \quad (13)$$

3.1.5 Nonparametric distributions

Notice that the above computational forms are applicable even in the case that there are no distributional assumptions. In particular, we apply the above CUSUM stopping rules (12), (13) by only assuming that $E[X_i] = \theta_0$ for $i = 1, 2, \dots, \tau - 1$ and $E[X_i] = \theta_1$ for $i = \tau, \tau + 1, \dots$

4. CUSUM in Video Sequences

To describe how this framework applies to video processing tasks, we use two indicative examples, namely object tracking and video segmentation. In both cases there is a need of change-point detection. For the case of tracking, we detect the time instances when a continuous tracker loses track, and thus we must re-initialize it. For the video segmentation case, we need to detect temporal boundaries or equivalently time (frame) windows of interest.

For the case of hand tracking, we primarily use the model-based continuous tracker of [9], to obtain hand configurations fast and accurately. In this case, the high *dofs*, the fast finger articulations, the large rotations and the frequent occlusions cause error accumulation, difficult to overcome, leading to the loss of track. Thus, our aim is to detect the time instances when the continuous tracker starts losing track, so that we can re-initialize it. As an example of

this re-initialization, we used the appearance-based discrete tracking method of [19]. Note that we use these two trackers as an example, and their detailed description is beyond the purposes of this paper.

An important task to achieve accurate tracking by coupling two different trackers, such as [9] and [19], is to define a criterion that will determine whether the continuous tracker performs well or may fail. Based on the mathematical formulation as described in 3.1, this criterion should play the role of the observations X_i 's, in which we aim to detect the change-points. For the 3D hand tracking application, as observations we use the differences between the hand contour in two successive frames. More specifically, let $\mathbf{C}_{i-1} = [C_{i-1}(p) \mid p = 1, \dots, P]$ be the extracted 2D hand contour on the image plane, at frame $i - 1$, for a pre-defined number of points P . Similarly, let $\mathbf{C}_i = [C_i(p) \mid p = 1, \dots, P]$ be the extracted hand contour at the next frame i . Obviously, since these contours are the result of a 2D tracking, there is correspondence between the contour point sets \mathbf{C}_{i-1} and \mathbf{C}_i .

We define as observation X_i for the CUSUM procedure and for each frame i , the difference between two successive hand contours,

$$X_i = \|\mathbf{C}_i - \mathbf{C}_{i-1}\| = \sqrt{\sum_{p=1}^P [C_i(p) - C_{i-1}(p)]^2} \quad (14)$$

Notice that the above distance between successive contours does not only denote how much the hand shape changes but also estimates the contour displacement on the image plane. In both abrupt displacements and shape changes, any continuous tracker may fail as explained before.

Fig. 1 illustrates an example of 3D hand tracking using the trackers of [9] and [19], in 450 frames of a sequence (seven key-frames are shown). The first column shows the original frame, the second column shows the continuous tracking ([9]) results, whereas the third column shows the discrete tracking ([19]) results. In frame 250 we can see that the continuous tracker starts losing track. In this frame, we need to re-initialize the continuous tracker, therefore we use the discrete tracker result. After the model re-initialization, we can see that the continuous tracker performs well.

The corresponding hand contour changes, used as observations X_i 's in the CUSUM procedure, are illustrated in the plot of Fig. 2, where the estimated change-point is the frame 247. The corresponding observation value (contour change) is $X_{247} = 6.1336$. In this plot we also show that the continuous tracker needs to be re-initialized at more than one frames. The window in which the discrete tracker gives solutions is defined by the horizontal line $X = 6.1336$ shown in the plot.



Figure 1. Continuous tracking results and tracker re-initialization (frame #250) from the discrete tracker.

For the case of video segmentation, and to illustrate the performance of our approach on specific segmentation criteria, we focus on American Sign Language (ASL) videos.

Most signs in American Sign Language (ASL) and other signed languages are articulated through the use of particular handshapes, orientations, locations of articulation relative to the body, and movements. However, a subclass

of signs in ASL, the fingerspelled signs - generally proper names and other borrowings from the spoken language - are instead produced by concatenating handshapes that correspond to the 26 letters of the alphabet [21].

Recognition strategies for fingerspelling [5] must be fundamentally different from those used for other signs. However, fingerspelling segmentation within a fluent stream of signing is non-trivial, as many of the same handshapes that

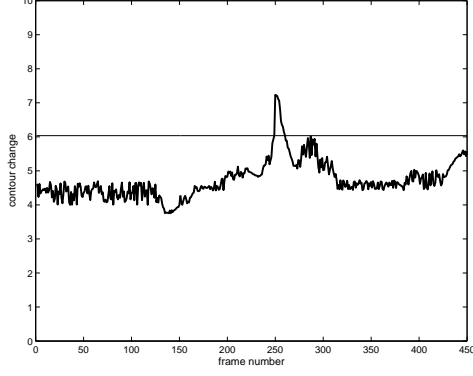


Figure 2. CUSUM results for the change-point detection in the case of Fig. 1.

are used as letters are also used in the formation of other types of signs.

In our framework, we exploit an identical property of fingerspelling to facilitate its segmentation, i.e., the more rapid movements of individual fingers that occur during this phase, than the finger movements that are typically found in other types of signs. Thus, we apply the CUSUM procedure to detect faster finger articulations, using the hand shape changes. Note that fast finger articulations correspond to great changes of the projected hand contour. On the other hand, the contour changes defined in eq. (14) cannot be used in this case, since we are only interested in shape changes and not the displacements of the hand.

We define as observation X_i for the CUSUM procedure and for each frame i , the difference between the curvatures of two successive hand contours C_{i-1} and C_i ,

$$X_i = \|\mathbf{K}_i - \mathbf{K}_{i-1}\| = \sqrt{\sum_{p=1}^P [K_i(p) - K_{i-1}(p)]^2}, \quad (15)$$

where $\mathbf{K}_{i-1} = [K_{i-1}(p) \mid p = 1, \dots, P]$ and $\mathbf{K}_i = [K_i(p) \mid p = 1, \dots, P]$ are the curvatures of the contours C_{i-1} and C_i respectively. Also, it is $\mathbf{K}_l = \frac{\dot{x}_l \ddot{y}_l - \dot{y}_l \ddot{x}_l}{[\dot{x}_l^2 + \dot{y}_l^2]^{3/2}}$, for $l = i-1, i$, where (x_l, y_l) are the cartesian coordinates of the contour on the image plane.

Table 1 shows the segmentation results for eight ASL videos. As mentioned in the introduction, the videos we used are annotated, so that we know what is said (signed) and we have as ground-truth the actual frames where fingerspelling is performed. The first column shows the video number, the second column shows the actual number of the fingerspelling segments in the sequence, the third column represents the ground-truth fingerspelling frame windows, and the fourth column shows the CUSUM results using the hand curvature change of eq. (15). The main reason for the difference between the actual and estimated boundaries is that before and after the actual fingerspelling, there is increased finger articulation, which does not correspond to the

<i>vid.</i>	<i>seg.</i>	<i>ground – truth</i>	<i>CUSUM</i>
(1)	1	(43 – 55)	(36 – 57)
(2)	1	(151 – 181)	(146 – 185)
(3)	1	(43 – 65)	(45 – 69)
(4)	1	(71 – 81)	(69 – 85)
(5)	2	(53 – 67, 87 – 101)	(55 – 71, 85 – 101)
(6)	2	(51 – 69, 83 – 101)	(46 – 73, 81 – 101)
(7)	2	(25 – 47, 145 – 173)	(21 – 45, 143 – 175)
(8)	2	(21 – 29, 125 – 159)	(19 – 31, 121 – 163)

Table 1. Fingerspelling segmentation results for eight ASL videos

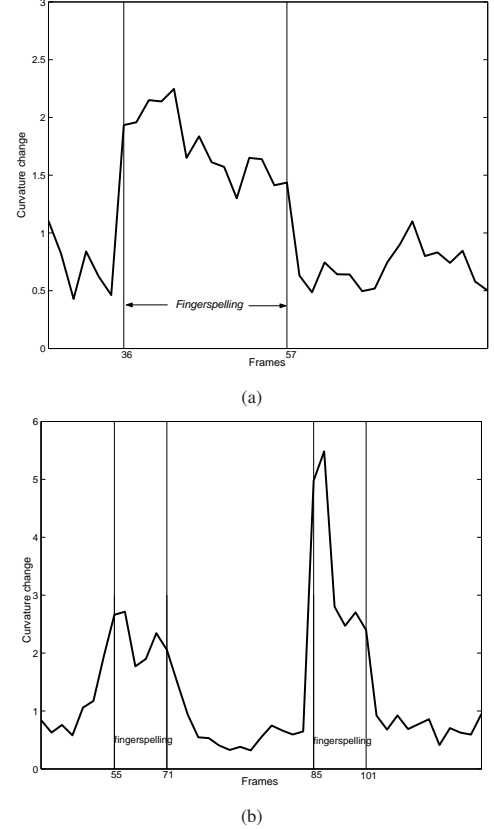


Figure 3. CUSUM results for the fingerspelling segmentation in videos (1) and (5) of Table 1.

fingerspelling phase, but it is just a transition to and from this phase.

Two graphical examples of the fingerspelling segmentation are shown in Fig. 3. Both plots represent the input observations as defined in eq. (15), and the estimated fingerspelling boundaries are shown with the lighter lines. In Fig. 3(a), we show the results for video (1) of Table 1, where there is only one fingerspelling phase. In Fig. 3(b), we show the results for video (5) of Table 1, where both fingerspelling segments are successfully detected and their boundaries are accurately estimated.

5. Conclusions

In this work, we presented the statistical CUSUM change-point detection procedure applied to video sequences to detect motion changes and switch between different methods. Our motivation came from problems where temporal continuity assumptions are violated. Two indicative examples of such problems is the continuous tracking of articulated objects and the video segmentation. In the first case, abrupt movements, fast articulations, large rotations and occlusions lead to abrupt changes that usually cannot be handled by continuous trackers. In the second case, although some of the existing dynamic classification methods perform well, abrupt changes in the observations need to be detected for switching between different models. The main advantages of our framework are: (i) it can detect changes online, (ii) it is robust to noisy observations, (iii) it does not require training, (iv) it is general enough to be used by recognition/classification methods, when training data are available and abrupt changes occur. We applied our framework to the case of continuous 3D hand tracking to re-initialize the tracker when it fails, using a discrete tracker. In this case, the high dofs, the fast finger articulations, the large rotations and the frequent occlusions cause error accumulation. Also we illustrated the performance of our approach in video segmentation, and specifically on segmentation of fingerspelling in American Sign Language (ASL) videos.

References

- [1] V. Athitsos and S. Sclaroff, "Database Indexing Methods for 3D Hand Pose Estimation," *Gesture Workshop*, Genova, Italy, April 2003. 2
- [2] G. Casella, and R.L. Berger, *Statistical Inference*, Duxbury Press, 1990. 3
- [3] B. Chen, and P. Willett, "Detection of Hidden Markov Model Transient Signals," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 36(4), October 2000. 1, 2
- [4] J.L. Devore, *Probability and Statistics for engineering and the sciences*, Pacific Grove, Calif.: Brooks/Cole Pub. Co, 2004. 1
- [5] K. Grobel and H. Hienz, "Video-based recognition of fingerspelling in real time," *Workshops Bildverarbeitung fr die Medizin*, Aachen, 1996. 2, 5
- [6] M. Isard, and A. Blake, "Contour Tracking by Stochastic Propagation of Conditional Density," *European Conference on Computer Vision*, Cambridge UK, 1996. 1, 2
- [7] J. Lin, Y. Wu and T.S. Huang, "Modeling the Constraints of Human Hand Motion," *5th Annual Federated Laboratory Symposium (ARL2001)*, Maryland, 2001. 2
- [8] G. Lorden, "Procedures for reacting to a change in distribution," *Annals of Mathematical Statistics*, vol. 42, pp. 1897-1908, 1971. 2, 3
- [9] S. Lu, D. Metaxas, D. Samaras and J. Oliensis, "Using Multiple Cues for Hand Tracking and Model Refinement," *IEEE Conference on Computer Vision and Pattern Recognition*, Wisconsin, June 2003. 2, 4
- [10] G.V. Moustakides, "Optimal stopping times for detecting changes in distributions," *Annals of Statistics*, vol. 14(4), pp. 1379-1387, 1986. 2, 3
- [11] E.S. Page, "A Test for a Change in a Parameter Occurring at an Unknown Point," *Biometrika*, vol. 42, pp. 523-526, 1955. 2
- [12] L.R. Rabiner, "A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77(2), pp. 257-286, 1989. 1
- [13] J. Reh and T. Kanade, "Model-based Tracking of Self-occluding Articulated Objects," *IEEE International Conference on Computer Vision*, Cambridge, MA, June, 1995. 2
- [14] A.N. Shiryaev, *Optimal Stopping Rules*, Springer-Verlag, 1978. 2
- [15] H. Sorenson, *Kalman Filtering: Theory and Application*, IEEE Press, 1985. 1, 2
- [16] B. Stenger, A. Thayananthan, P.H.S. Torr and R. Cipolla, "Hand Pose Estimation Using Hierarchical Detection," *International Workshop on Human-Computer Interaction*, Prague, Czech Republic, May 2004. 2
- [17] E.B. Sudderth, M.I. Mandel, W.T. Freeman, and A.S. Willsky, "Visual Hand Tracking Using Nonparametric Belief Propagation," *IEEE CVPR Workshop on Generative Model Based Vision*, Washington DC, May 2004. 2
- [18] C. Tomasi, S. Petrov and A. Sastry, "3D Tracking = Classification + Interpolation," *IEEE International Conference on Computer Vision*, Nice, France, October 2003. 2
- [19] G. Tsechpenakis, D. Metaxas, and C. Neidle, "Learning-based Dynamic Coupling of Discrete and Continuous Trackers," *IEEE ICCV Workshop on modeling People and Human Interaction (PHI'05)*, Beijing, China, October 2005. 1, 4
- [20] C. Vogler and D. Metaxas, "A Framework for Recognizing the Simultaneous Aspects of American Sign Language," *Computer Vision and Image Understanding*, 81, pp. 358-384, 2001. 2
- [21] S. Wilcox, "The Phonetics of Fingerspelling," *Studies in Speech Pathology and Clinical Linguistics*, 4, Amsterdam and Philadelphia: John Benjamins Publishing Co., 1992. 5
- [22] H. Zhou and T.S. Huang, "Tracking Articulated Hand Motion with Eigen Dynamics Analysis," *IEEE International Conference on Computer Vision*, Nice, France, October 2003. 2