2011 Seventh International Conference on Signal Image Technology & Internet-Based Systems

# Hand gesture modeling using Dynamic Bayesian Networks and Deformable Templates

Artur Wilkowski
Faculty of Geodesy and Cartography
Warsaw University of Technology
Pl. Politechniki 1, 00-661 Warsaw, Poland
Email: A.Wilkowski@gik.pw.edu.pl

Włodzimierz Kasprzak
Institute of Control and Computation Engineering
Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland
Email: W.Kasprzak@elka.pw.edu.pl

***Abstract*—The paper presents a stochastic approach to articulated hand (palm shape) tracking in images. The gesture model is given in terms of a Dynamic Bayesian network that incorporates a Hidden Markov Model in order to utilize prior information on gesture structure in the tracking task. The Deformable Templates methodology is applied for hand shape modeling. Experimental evaluation of articulated hand tracking in cluttered environment using particle filtering is provided. A comparison of this method with a typical tracking approach, that makes no use of temporal gesture information, is also given.**

## I. Introduction

Hand gesture recognition in images [1], [2], [3], [4] is an important issue in the Human-Machine Interaction domain. Gestures may carry commands and some additional information, e.g about the object of action related to a command, which can be of paramount importance for instance in robot control applications. Under the term "hand gesture" we shall understand here a meaningful sequence of palm postures and/or palm positions.

The probabilistic modeling has been so far the most successful way of modeling dynamic gestures. As a generic probabilistic model, the Hidden Markov Model (HMM) [5] was typically chosen for gesture representation. The HMM models interdependent stochastic processes, one hidden and one observed. The hidden process (modeled as Markov Chain) can be viewed as representing the temporal structure of the gesture, the observable process (dependent on the hidden one) represents hand articulation and motion over time. A common usage of the Hidden Markov Model of gesture is in the task of gesture recognition. The processing scheme typically follows the bottom-up processing mode: the low level image processing is used first to detect the hand and its features in every frame, and then a Hidden Markov Model-based evaluation of such sequence of features follows in order to perform gesture recognition [1], [4], [6].

However, not much is known about utilization of the powerful knowledge on gesture articulation, that is stored in a Hidden Markov Model, for improving hand detection in images and the estimation of hand parameters. The utilization of a HMM model is not straightforward in the tracking task, since HMM lack capabilities of modeling auto-regressive processes. Nevertheless, the HMM model can still become a part of more complex probabilistic framework such as the Switched Kalman Filter [7] or the Dynamic Bayesian Networks (DBN) [8].

The application of DBN to gesture modeling has not been widely discussed so far. In one early solution [9] an extended version of the Switched Kalman Filter was applied for gesture tracking. However, the approach concentrated on motion modeling while the appearance model was reduced to hand representation as an ellipsoid image patch. In [10] the discreet state Bayesian network was used as a gesture model, and the particle filter was used to perform stochastic inference. In [11] the DBN formalism was used for the solution of a two-hand gesture recognition problem.

Some other solutions that do not explicitly use the DBN formalism but are related in terms of adopting the concept of exploiting gesture models to support the task of hand tracking have also been devised. In [12] so called joint Bayesian framework is proposed that consists of the particle filter tracker of hand position cooperating with the discreet HMM used for generating filtered estimates of hand posture. In the hand tracking system given in [13] the modes of dynamic hand motion are subject to change due to different values of a discreet variable. In related approach [14] the values of some discreet variable select the right or left hand for tracking. In [15] the space of rigid and articulated hand motion is made discreet and a deterministic hierarchical tracker is applied.

Another problem related to the concept of gesture tracking concerns the selection of a generative palm model (either 2D or 3D) capable of representing arbitrary hand postures.

The main contribution of this paper is the proposition of an integrated stochastic model for tracking articulated hand (palm) motion, designed in form of a Dynamic Bayesian Networks. The discrete part of this model is explained in terms of Hidden Markov Model - it is the main information source on expected hand gestures and sequences of gestures. A second part (with discrete and continuous variables) connects the model with the Deformable Templates method for shape modeling and model-to-image matching [16], [17]. We will show that our model can be successfully used for articulated hand tracking in cluttered image conditions and that the utilization of the prior knowledge on gesture structure increases the robustness of tracking results.

The paper is organized as follows: in sec. II there is a

978-0-7695-4635-3/11 $26.00 © 2011 IEEE
DOI 10.1109/SITIS.2011.55

hand shape modeling approach presented that is based on the Deformable Templates method, in sec. III our gesture model (a DBN) is introduced, the hand tracking algorithm is explained in sec. IV and experimental results are given in sec. V

## II. Shape modeling with Deformable Templates

In our earlier work [17] the Deformable Templates were used to perform static hand pose recognition. Now, we will use them for 2D palm modeling in the context of Bayesian gesture representation. A 1-dimensional curve $x(s)$ is constructed as a weighted combination of $N_b$ basis functions, which we will denote as $B_n(s)$, where $n = 0, ..., N_b - 1$; and $s$ is the spline parameter. Thus:

$$x(s) = \sum_{n=0}^{N_b-1} x_n B_n(s), \tag{1}$$

where $x_n$ is the value of the n-th control point and $B_n(s)$ is the value of the n-th spline basis function at point $s$. A compact matrix notation of the curve would be:

$$x(s) = \mathbf{B}(s)^T \mathbf{Q^x}, \tag{2}$$

$$\mathbf{B}(s) = \begin{pmatrix} B_0(s) \\ ... \\ B_{N_b-1}(s) \end{pmatrix}; \quad \mathbf{Q^x} = \begin{pmatrix} x_0 \\ ... \\ x_{N_b-1} \end{pmatrix}$$

where $\mathbf{B}(s)$ is a vector of values for all basis functions and $\mathbf{Q^x}$ denotes a vector of point coordinates.

A 2-dimensional spline is a composition of two independent 1-dimensional splines founded on the same spline basis:

$$\mathbf{r}(s) = (x(s), y(s)) \tag{3}$$

A compact matrix notation of the 2-dimensional spline is:

$$\mathbf{r}(s) = \mathbf{U}(s)\mathbf{Q}, \tag{4}$$

where $\mathbf{r}(s)$ is a vector $(x, y)$ of spline coordinates for the given spline parameter $s$, $\mathbf{U}(s)$ is a matrix defined as

$$\mathbf{U}(s) = \begin{pmatrix} \mathbf{B}(s)^T & \mathbf{0} \\ \mathbf{0} & \mathbf{B}(s)^T \end{pmatrix}; \quad \mathbf{Q} = \begin{pmatrix} \mathbf{Q^x} \\ \mathbf{Q^y} \end{pmatrix}$$

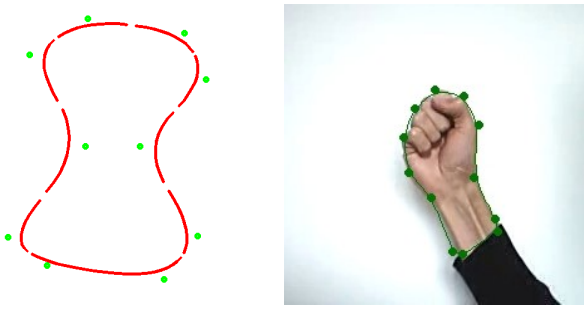
and $\mathbf{Q}$ is a double-length control vector made up of individual control vectors for the x- and y-dimension.

The resulting spline curve tries to approximate a polygon made up of values of control points (e.g. Fig. 1a). See Fig. 1b for quadratic periodic spline approximation of a fist outline.

In order to calculate similarities between curves the $L_2$ norm for a 2-dimensional curve is defined as:

$$\|\mathbf{r}\|^2 = \frac{1}{L} \int_{s=0}^{L} |\mathbf{r}(s)|^2 ds \tag{5}$$

The spline curve typically has many degrees of freedom (the splinespace has large dimensionality). In order to perform robust matching of the contour against image features it is desirable that some constraints would be imposed on the space of allowed shapes. In [16] a so called *shapespace* is used to reduce the number of parameters describing the shape. Each



(a) (b)

Fig. 1. B-spline examples: (a) periodic B-spline and control points follows the polygon outlined by the set of subsequent control points; (b) B-spline approximation of sample hand posture

point in the shapespace is denoted by $\mathbf{X}$. To establish a relation between the shapespace and the splinespace we need two parameters: the template spline $\mathbf{Q_0}$ and the weight matrix $\mathbf{W}$, expressing how a shapespace point $\mathbf{X}$ influences the shape and position of the resulting spline. The relation can thus be described by the equation:

$$\mathbf{Q} = \mathbf{WX} + \mathbf{Q_0}, \tag{6}$$

where $\mathbf{Q_0}$ is the original template shape and $\mathbf{Q}$ is the shape after transformation.

There are infinitely many possibilities of defining the shapespace. One of the simplest examples of the shapespace is the space of Euclidean transformations, comprised of template shape rotation, translation and scaling. The transformation matrix in the shapespace of Euclidean similarities is specified as follows:

$$\mathbf{W} = \begin{pmatrix} \mathbf{1} & \mathbf{0} & \mathbf{Q_0^x} & \mathbf{-Q_0^y} \\ \mathbf{0} & \mathbf{1} & \mathbf{Q_0^y} & \mathbf{Q_0^x} \end{pmatrix} \tag{7}$$

The values of the 4-element vector in the *shapespace* of Euclidean similarities can be interpreted as:

$$\mathbf{X}_n = [c_x, c_y, k\cos\theta - 1, k\sin\theta]^T,$$

where $c_x, c_y$ are responsible for translation, $\theta$ is the rotation angle and $k$ is the scaling coefficient.

In order to measure similarity between two curves specified by shapespace data a distance metric is introduced (using metric matrix $\mathcal{U}$) that applies the $L_2$ norm of the spline, specified in (5), as:

$$\|\mathbf{X}\| = \sqrt{\mathbf{X}^T\mathbf{W}^T\mathcal{U}\mathbf{WX}} \tag{8}$$

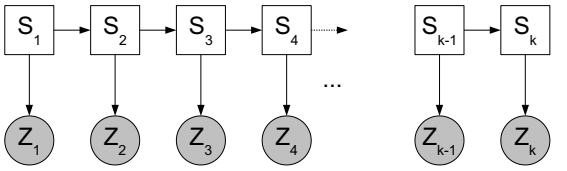
The norm of the vector $\mathbf{X}$ in shapespace is the distance of the corresponding spline from the template spline:

$$\|\mathbf{X}\| = \|\mathbf{X} - \mathbf{0}\| = \|\mathbf{Q} - \mathbf{Q_0}\|$$

## III. Stochastic Gesture Model

### A. Dynamic Bayesian Networks

At its simplest form the Bayesian Network (BN) can be regarded as a graphical model of a probabilistic system that




(a) "Unrolled" example of DBN (b) 2TBN example of DBN

Fig. 2. Different representations of a DBN

represents *conditional independencies* between variables [8]. The Bayesian network represents some particular factorization of the joint distribution. Random variables correspond to nodes in a graph and a directed arc from node $A$ to $B$ denotes that $B$ is (directly) conditioned on $A$. Each Bayesian Network can be fully described using conditional probability density functions (CPD) of its variables.

A Dynamic Bayesian Network (DBN) allows to model probability distributions over semi-infinite collections of random variables ( $\mathbf{V}_1 = \{V_1^1, ..., V_1^N\}, \mathbf{V}_2, ...$) or in other words - to design models of discrete time stochastic processes. The DBN can be naively represented using the formulation of the Bayesian Network, where each of the series of random variables ($\mathbf{V}_1, ..., \mathbf{V}_i$, $i = 1, 2, ...$) is treated separately (see fig. 2a, where $\mathbf{V}_i = \{S_i, Z_i\}$ ). If we assume however that the model is a first-order Markovian one (higher orders are also possible) then the DBN can be represented more compactly as a pair $(B_1, B_\rightarrow)$, where $B_1$ is a BN defining the prior $P(\mathbf{V}_1)$, and $B_\rightarrow$ is a two-slice temporal Bayes Network (2TBN) which describes the transition model $P(\mathbf{V}_k|\mathbf{V}_{k-1})$. The transition probability can be factorized as:

$$P(\mathbf{V}_k|\mathbf{V}_{k-1}) = \prod_{i=1}^{N} P(V_k^i|Pa(V_k^i)) \tag{9}$$

where $Pa(V_k^i)$ are parents of the node $V_k^i$. Therefore it turns out that under first order Markov assumption only two time slices are required to describe the dynamics of the system. Most often it is assumed that the transition model is time invariant, so $P(\mathbf{V}_k|\mathbf{V}_{k-1})$ is constant for arbitrary $k$. An example of 2TBN representation is given in Fig. 2b. Given a finite number of time-steps, a 2TBN can be easily "unrolled" into corresponding DBN, which have the form of a classical BN. The joint distribution for such an "unrolled" network is:

$$P(\mathbf{V}_{1:K}) = \prod_{i=1}^{N} P(V_1^i) \prod_{k=2}^{K} \prod_{i=1}^{N} P(V_k^i|Pa(V_k^i)) \tag{10}$$

### B. DBN structure for gesture modeling

In (Fig. 3) our DBN model is presented, designed for tracking and recognition of gestures in images. The model can be divided into three distinctive segments. The first segment of discreet variables $M_k$, $H_k$, $F_k$ and $Y_k$ covers the selection

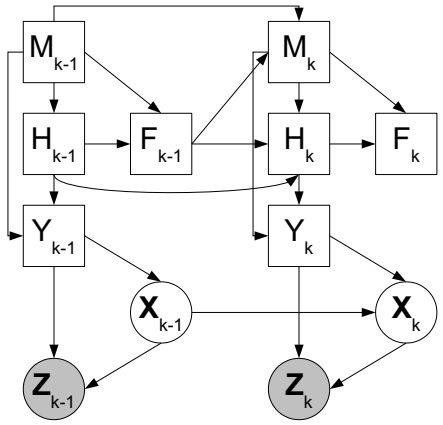



Fig. 3. DBN Model for hand tracking and gesture recognition

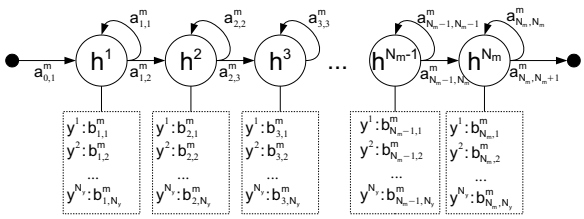



Fig. 4. Structure of a single-word HMM model $m^i$ example as a left-right HMM (the index $i$ is dropped for convenience)

of an appropriate gesture model, a particular palm shape and a motion model. An analogy between the Hidden Markov Model and the discreet part of our network is explained in the subsequent sections. The second segment contains continuous variables $\mathbf{X}_k$ and it controls the object's dynamics in 3-D space, founded on the concept of a linear dynamic system (LDS). The third segment contains the only observable variable $\mathbf{Z}_k$ and it represents measurements in the spline-space.

### C. CPDs of the discreet variables

*1) Single gesture:* The representation of a single gesture (also called a word from now on) can be easier explained if the appropriate part of our 2TBN, with discrete variables $H_k$, $Y_k$, and $F_k$ is converted to an equivalent "left-to-right" Hidden Markov Model. The "state" variable is $H_k$, the "output" variable is $Y_k$, and the variable $F_k$ plays the role of additional transition conditions. A structure of this HMM for a single gesture is given in Fig. 4 (it uses HMM states and not the Bayesian notation!).

The possible values of variable $H_k$ in our DBN correspond to "hidden" states in the Hidden Markov Model. The transition pdf of this variable is made explicit due to state transitions in the HMM. As these states are organized (nearly) sequentially (i.e. a left-right model with loops or the Bakis model), a single

value of $H_k$ has the meaning of a particular *stage* of a single gesture being performed. By means of the hidden states we model the time-alignment of a gesture and can perform *time-warping* during gesture recognition. The state transition values depend on word (gesture) (represented by the variable $H_k$), on its own value at the previous stage and on the information whether the current word is complete or not (i.e. whether the variable $F_{k-1}$ takes its "final" value):

$$P(H_k = h_k | M_k = m_k, H_{k-1} = h_{k-1}, F_{k-1} = f_{k-1}) \\ = f_{k-1} a^{m_k}_{0,h_k} + (1 - f_{k-1}) a^{m_k}_{h_{k-1},h_k} \quad (11)$$

where $a^m_{i,j}$ are inter-state transition probability values (like in the HMM). Beside states representing values of the $H_k$ variable, there are two "virtual", non-emitting states - denoted under indexes: 0 (initial state) and $N_m + 1$ (final state for the given word model)), that act as a convenient "glue" between different word models. The additional transition probabilities, $a^m_{i,0}$ and $a^m_{i,N_m+1}$, are initial probabilities of HMM states and probabilities of transition to the final state (actually probabilities of setting the $F_k$ variable), respectively. The variable $Y_k$ corresponds to the outputs (emitted values, observations) of the HMM, however in our approach this variable is unobserved. This variable is the only discreet variable influencing the actual observation $Z_k$, and the continuous variable $\mathbf{X}_k$. The variable $Y_k$ is treated as an 'output' of the discreet part of the network giving information on the hand shape and its motion, expected at given stage of currently performed gesture. Firstly, it selects the shapespace for the observed shape. Secondly, it selects the motion model that controls the behavior of shape parameters. The variable $Y_k$ depends both on the "stage" variable $H_k$ and the "gesture" variable $M_k$:

$$P(Y_k = y_k | H_k = h_k, M_k = m_k) = b^{m_k}_{h_k,y_k} \quad (12)$$

The already mentioned variable $F_k$ is a convenience binary variable – when the word (gesture) is expected to be in its final stage the variable is set to 1, otherwise it is equal to 0. $F_k$ serves two purposes: 1) it helps to organize the inter-word-model connections without introducing complex CPDs between variables and 2) the probability assigned to this variable during inference provides the prior for the likelihood that the system could be found in the terminal state (of the word model) in the next time step. The conditional probability distribution of the $F_k$ variable is given by the subsequent conditional probability distribution:

$$P(F_k = f_k | H_k = h_k, M_k = m_k) = a^{m_k}_{h_k,N_{m_k}+1}$$

*2) Set of gestures:* The HMMs, each representing a single gesture, are parts of a larger HMM model representing sequences of such gestures. The single gesture HMMs are alternative paths, closed in a loop-like fashion. This enables to recognize a stream of gestures and utilize the a-priori Markovian knowledge on word (gesture) succession probabilities (coming for instance from analysis of a sentence grammar). The variable $M_k$ represents the current *word* (or a specific gesture) and the intra-model variables ($H_k$, $Y_k$ and $F_k$) are conditioned on $M_k$. The structure of inter-connected HMMs for gesture sequence recognition is given in Fig. 5.

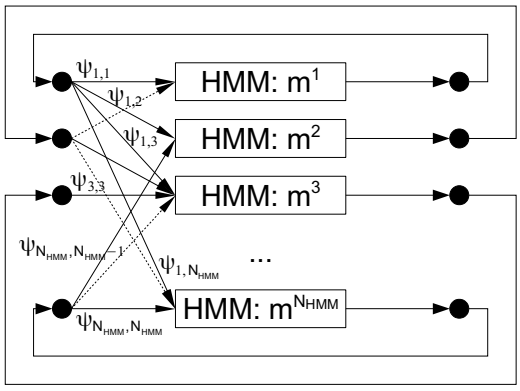



Fig. 5. Structure of inter-word-model connections in the network

The variable depends on its own value in the previous time step and also on the information whether the current word model reached its final state (in the previous time step), which is modeled by the variable $F_k$:

$$P(M_k = m_k | M_{k-1} = m_{k-1}, F_{k-1} = f_{k-1}) = \\ \begin{cases} (1 - f_{k-1}) + f_{k-1} \psi_{m_{k-1},m_k} & \text{if } m_{k-1} = m_k \\ f_{k-1} \psi_{m_{k-1},m_k} & \text{if } m_{k-1} \neq m_k \end{cases} \quad (13)$$

The CPD of inter-word transitions is generated by some arbitrary transition $\psi_{i,j}$ function. The $\psi_{i,j}$ function models the probabilities of transitions between different gesture models (words) $m^i$ and $m^j$. The function takes only 2 arguments and is able to capture only first-order Markov relations between words.

### *D. CPD of the observation*

In the ideal case the observation in our system is the hand shape described by a spline curve. The observation depends on the continuous variable $\mathbf{X}_k$ and the discreet variable $Y_k$. The observation equation is directly derived from the shapespace-to-splinespace projection equation (6) – it has the form:

$$\mathbf{Q}_{z,k} = \mathbf{W}^{y_k} \mathbf{X_k} + \mathbf{Q}^{y_k}_0, \quad (14)$$

where both the weight matrix $\mathbf{W}^{y_k}$ and the template $\mathbf{Q}^{y_k}_0$ depend on the output $y_k$ of the HMM, i.e. a value of the variable $Y_k$. Thus, the state $y_k$ is actually a selector of different observation models - by changing $\mathbf{W}$ and $\mathbf{Q}$ it enables to interpret the values of variable $\mathbf{X}$ as different shapes.

### *E. CPD of the $\mathbf{X}_k$ variable*

The variable $\mathbf{X}_k$ represents the palm shape in terms of an unobservable state of a dynamic system. $\mathbf{X}_k$ is conditioned on its value from the previous time step $\mathbf{X}_{k-1}$ and on the discreet output of the "gesture" HMM, $Y_k$. The variable describes such continuous properties of the palm object like position, size

and orientation, but it also may represent deformations of the object's shape, modeled in the shapespace.

The general model of dynamics of the $\mathbf{X}_k$ variable, $P(\mathbf{X_k}|\mathbf{X_{k-1}} = \mathbf{x}_{k-1}, Y_k = y_k)$, is chosen to be a first-order auto-regressive process, which can be most simply expressed in generative form [16]:

$$\mathbf{X}_k = \mathbf{T}^{y_k}\mathbf{X_k} + \mathbf{d}^{y_k} + \mathbf{u}_k \tag{15}$$

where $\mathbf{T}^{y_k}$ is a matrix defining a continuous state transition model, $\mathbf{u}_k$ is a zero-mean Gaussian noise with covariance matrix $\mathbf{R}^{y_k}$, $\mathbf{u}_k \sim \mathcal{N}(0, \mathbf{R}^{y_k})$, and $\mathbf{d}^{y_k}$ is a vector of constant drift.

The above dynamic model is rather general and in our solution it can represent different sorts of shape transformations. E.g. it can represent a set of rigid transformations (such as Euclidean Similarities) or non-rigid ones (where the space of shapes is spanned over some set of so called *key-frames* [16]). Although several experiments were performed with both these transformations, in this paper we will concentrate only on the first case described in sec. II. Here the transition matrix is set to identity, $\mathbf{T}^{y_k} = \mathbf{I}$, and the noise covariance matrix is:

$$\mathbf{R}_k = \sigma_0(\mathcal{H}^{y_k})^{-1}$$

and $\mathcal{H}^{y_k} = (\mathbf{W}^{y_k})^T\mathcal{U}^{y_k}\mathbf{W}^{y_k}$. With such defined parameters we get:

$$P(\mathbf{X}_k|\mathbf{X}_{k-1}) \propto \exp(-\frac{1}{2\sigma_0^2}\|\mathbf{X}_k - \mathbf{X}_{k-1} - \mathbf{d}^{y_k}\|^2) \tag{16}$$

Thus, the dynamic model is an isotropic *random walk with drift*, defined in such a way that the specific value of probability density corresponds to a fixed distance from the previous curve $\mathbf{X}_{k-1}$ plus the constant drift vector $\mathbf{d}^{y_k}$ - of course, the larger the distance, the smaller the value of probability density. Intuitively the dynamic model should be understood as allowing the movements from the space of Euclidean Transformations (translation, rotation, scaling), but the movements, except for translation component, are probabilistically constrained to lie near the previous curve with respect to the $L_2$ norm. In such a case the palm shape is treated as a rigid object and all the gesture articulation must be encoded by the observation variable $Y_k$ - allowing to 'switch' shapes according to the $y_k$ value. If the space of hand poses is densely quantized or if motion between different hand poses is rapid enough to neglect transient hand postures, it can well approximate the real hand dynamics. Since it may be very easy defined 'by-hand' the Euclidean Transformation model is used in this work for the evaluation of inference methods.

## IV. Hand tracking

### A. The inference task in DBN

Our gesture model allows to interpret such inference tasks in DBN, like *filtering* and *most likely explanation* [8], as hand tracking and gesture recognition processes. The tracking problem is regarded as a "state" variable filtering task - its goal is to find the posterior joint distribution of all the "state" variables in the network, $\mathbf{V}_k = (\mathbf{X}_k, M_k, H_k, F_k, Y_k)$, given evidences (observation sequence of output variables). At time step $k$ the sequence of observations $z_{1:k}$ is available and the filtering task is to estimate the probability $P(\mathbf{V}_k|z_{1:k})$. Although our DBN is suitable for a Switched Kalman Filter (under some assumptions), we present results obtained with the Particle Filter [18], which is a straightforward solution for approximating multimodal distributions and is expected to be more resistant to image clutter.

In the Particle Filtering approach the posterior distribution is approximated by a set of (weighed) particles. We apply a modified version of the particle filter, called *Rao-Blackwellized* Particle Filter (RBPF) [8], where only the continuous variables were sampled while the discreet ones were integrated analytically.

The gesture recognition problem may be solved by a *token passing* search [19], a modification of the well known *Viterbi search* [5], performed for the overall HMM, In the experiments we shall concentrate on using our DBN for palm tracking, and especially verifying the use of high-level information (about the gesture) for shape and motion prediction.

### B. Image preprocessing

The experiments were performed on edge images obtained from grayscale images. A grayscale image was obtained due to a transformation of the color image acquired by a camera. The input image was first smoothed and converted into grayscale, then an edge image was generated by the Sobel operator, and afterwards an edge thinning operator removed the least significant edges. The edges detected correspond both to the palm and hand outline, to within-palm lines (such as lines between fingers), as well as to clutter from other objects in camera's field of view. Example of a preprocessed edge image is given in Fig. 8b.

### C. Practical measurement process

Although the theoretic measurement model was given sec. III-D it is not realistic mainly due to the fact that it is hard to obtain spline curve directly from the image. However, in case of particle filtering the only thing we must obtain during inference is the likelihood of the observation given some predicted spline. Let us consider the spline function $\mathbf{r}(s)$ associated with one particle at given timestep

$$\mathbf{r}(s) = \mathbf{U}(s)\mathbf{Q}$$

Now let us assume that the spline is projected onto the image and measurements are collected along the vectors normal distributed along the curve $\mathbf{r}(s)$ (measurements are simply the detected characteristic points such as edges). We will denote the observations taken along $j$-th normal as $z^j$. The observation likelihood for a single normal vector can be represented as [16]:

$$P(\mathbf{z}^j|\mathbf{r}(s_j)) \propto 1 + \frac{1}{\sqrt{2\pi}\sigma\alpha}\sum_t \exp(-\frac{(\nu_t^j)^2}{2\sigma^2}) \tag{17}$$

$\sigma$ and $\alpha$ are observation model parameters, $\mathbf{z}^j$ is a vector of observations along a single normal to the spline and

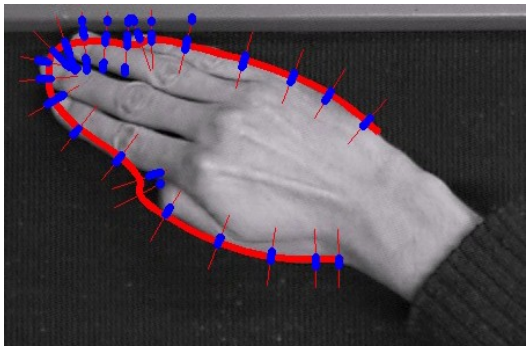


Fig. 6. Graphical representation of measurement model for particle filter

$\nu_t^j = |\mathbf{r}(s_j) - z_t^j|$ is interpreted as a distance between the point on the spline and the measurement. This model takes into account both the situations when several measurements were found along the normal (from which only one is true) or no observation was found (e.g. the object could be obscured). Finally, a strong assumption is made: the observations along different normals of the same spline can be considered independent (which may not be true in general). Hence, the total observation model is:

$$P(\mathbf{Z}_k|\mathbf{X}_k, Y_k) \propto \prod_j p(\mathbf{z}^j|\mathbf{r}(s_j))$$

## V. Experiments

### A. Experimental setup

For evaluation of inference methods in our DBN network the evaluation set of gestures given in Fig. 7 has been prepared and the DBN network distributions were set accordingly. The set of gestures consists of 4 different key hand postures (finger pointing upwards, thumb 'ok' sign, open palm and open palm with protruding thumb). The first two gestures are composed of 4 different static hand poses. Remaining two gestures each use a single hand posture (finger pointing upward) and they are dynamic – they start in static positions, in one of them the hand is moved to the right and then to the left, for the second gesture it is the opposite. It should be noted that the gesture set poses significant difficulties to any tracker/recognizer since the gestures are highly similar (e.g. they all start from the same hand pose and the first gestures end also in the same poses, while dynamic versions differ only by associated motion model).

For the experiments the probability distributions in the network were selected manually. A discreet part of the network was treated as a parallel composition of left-right Hidden Markov Models with loops (as shown in Fig. 4). Each submodel (corresponding to a single gesture) consisted of three emitting states and the probability of transition to the next state was arbitrarily set to $0.1$. For the static gestures the emission symbols consisted of numbers $0-3$ corresponding to four different hand postures modeled in the network. The

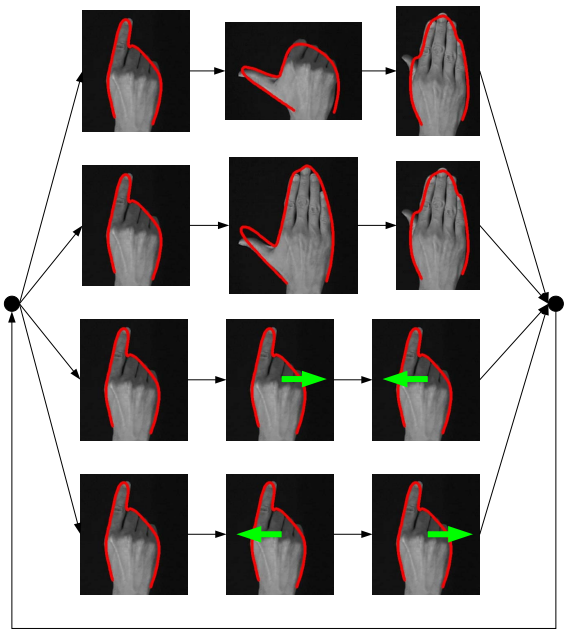


Fig. 7. Evaluation sequence for DBN with the Euclidean Dynamic Model

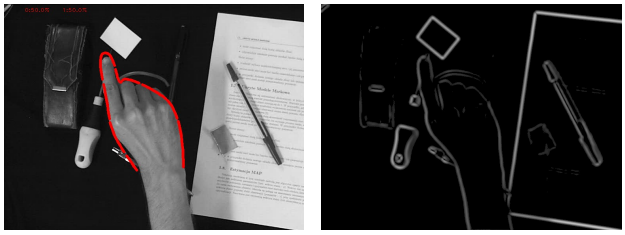
probability of emission of the 'correct' hand posture corresponding to the given state (as shown in 7) was set to $0.7$ while probabilities of remaining hand posture symbols were set to $0.1$ each. The addition of this small 'jitter' was intended to give better resistance of the tracking to occasional mismatches of hand postures.

In case of dynamic gestures additional output symbols $4, 5$ were used. Thus the symbols $0-3$ describe four static hand postures, while symbols $4$ and $5$ correspond to the hand posture 'finger pointing upwards' with associated dynamic model of movement to the left and to the right-hand side. For the dynamic gestures the probability of emitting of a 'correct' hand posture was set to $1.0$. The parameters of the dynamic model $P(\mathbf{X}_k|\mathbf{X}_{k-1}, Y_k)$ were set manually. The parameters of the observation model $P(\mathbf{Z}_k|\mathbf{X}_k, Y_k)$ were learned by first manually outlining each of hand posture contour manually to obtain a template curve $\mathbf{Q}_0$, aligning splines obtained in this way in order to make natural transitions between different hand postures, and computing remaining parameters (especially the matrix $\mathbf{W}$) to obtain the space of Euclidean Transformations.

Input for the tracker was the sequence of images captured from the camera. In order to evaluate robustness of approach the images were captured in grayscale and a significant amount of clutter was present in the images. Example of the input image is given in Fig. 8a.

### B. Results

*1) Gesture tracking:* In order to evaluate tracking performance for both static and dynamic gestures, 10 movie sequences were recorded each containing a continuous sequence of four gestures (giving 40 gestures in total), and the RBPF



(a) (b)

Fig. 8. Cluttered desktop environment used for evaluating tracker performance. (a) input grayscale image (b) edge image - the source of image observations

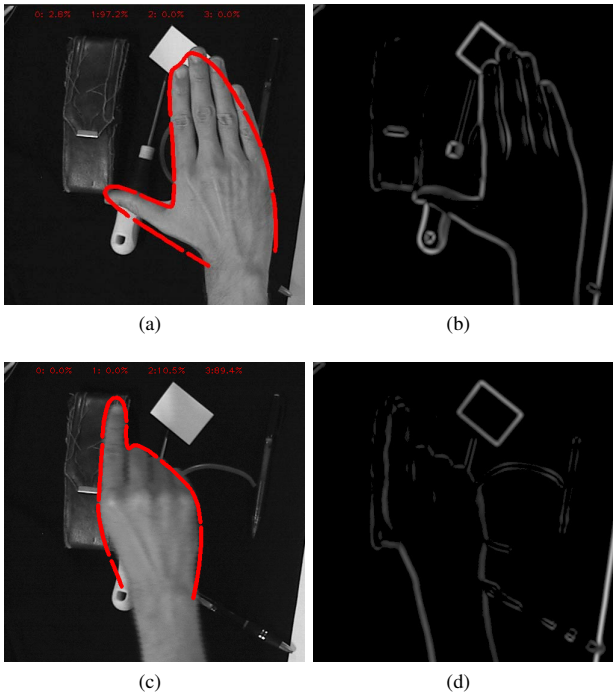
was used to perform tracking of the hand performing gestures. The experiments were performed in cluttered environment.

For the sake of experiment, it was assumed that the hand starts from a fixed position. During the test there were evaluated correctness of hand position, proper selection of the gesture model (denoted by the posterior marginal distribution of the $M_k$ variable), and proper selection of the hand posture - however in the latter case occasional mismatches were accepted. The number of only 100 particles was selected in order to simulate the on-line tracking.

The tracker was able to successfully track all the recorded sequences, with respect to all conditions defined above. One sequence required restart for the correct tracking (which is due to quite small number of particles used). Some examples from articulated hand tracking are given in Fig. 9. The conclusions from other, less formalized experiments, are that the system proposed is robust to clutter condition and it can also track rapid motion providing that it is consistent with the model of dynamics associated with specific gestures. The most important problem recognized were the imperfection in shape models, which led to occasional mismatches of hand postures prominent especially during transitions between different hand postures (before continuous parameters could be adjusted the their optimal value).

*2) Evaluation of HMM-guided hand tracking:* In the following section we will try to evaluate, to what extent the prior knowledge encoded in the HMM part of the Dymamic Bayesian Network may influence the hand tracking performance.

To see how the knowledge on hand articulation influences the tracking performance the DBN instance learned only to track the first two static gestures from the evaluation set were confronted with a simple artificial DBNs for which each symbol $Y_k$ (associated with single hand posture) was generated with equal probability (thus presenting no preference concerning hand shape). During the test no significant difference in performance between the two network instances was noted concerning hand postures recognized. In other words the network with prior knowledge on hand posture succession performed as well as the one with no such knowledge. The conclusion is that given a good approximation of hand position and effective matching method using Deformable Templates



(a) (b)

(c) (d)

Fig. 9. Examples of articulated hand tracking using our DBN definition and RBFP. (a) appearance of the open palm with protruding thumb is characteristic to the gesture model '1' and causes the probability mass to be concentrated into this model, (b) source of observations for matching in a), (c) rapid movement of the hand to the left side after static phase, causes the selection of corresponding 3rd gesture model, (d) source of observations for matching in c)

as well as limited number of postures to match, the prior knowledge is not actually required for differentiation between the given four hand postures.

This result is to some extent connected with the inference method used. In our RB particle filtering implementation the discreet states (which are responsible for hand posture selection) are not sampled. So the prior knowledge has no chance to direct search into areas of larger probability. The situation is expected to be different in case of other inference techniques such as Particle Filter with simple prior proposal.

In the next set of experiments, the performance of the DBN model designed for tracking gestures from the evaluation set, was compared against some artificial model on 10 test sequences containing only the two dynamic gestures recorded in cluttered environment. The goal was to evaluate to what extent the prior knowledge on expected motion influence the quality of tracking.

In our DBN the three switching dynamic models were utilized (stationary, motion to the left, motion to the right) with drift element initialized to $\mathbf{d}^{y_k} = [\pm 30, 0, 0, 0]^T$. The stochastic component of the motion was set to $\sigma_0 = 20$. Our main DBN instance was able to correctly track hand motion in 9 out of 10 sequences (in the remaining one tracking results depended on the specific run). The number of particles in the particle filter was set to 100.

The first comparative test was aimed at checking whether it is not sufficient to keep a single motion model for all gestures to obtain good tracking efficiency. In order to check this the DBN instance devised for tracking only the first two static gestures was tested on the same set of dynamic gesture sequences. In a single dynamic model the drift vector was set to 0 and the stochastic component was identical as above. From the set of 10 sequences only 4 (with most moderate movement) were correctly tracked (without the loss of track). Subsequently the stochastic component $\sigma_0$ was increased twofold, to see if this could make up for lower diversity of motion models. This time the track was kept to in 8 out of 10 sequences, however, during tracking serious discrepancies between real and recognized shape could be often spot.

The goal of the second comparative test was to see whether the knowledge of the sequence of motion model switching characteristic to each gesture brings value to gesture tracking as compared to the system with equal probabilities set to all models. For the experiments an artificial DBN specification was prepared, in such way that the new network was accepting the same symbols $Y_k$ as the DBN network designed for tracking in the full evaluation set, with exception that all the symbols had equal probability $\approx 1/6$. The dynamic model parameters were set just as in the first experiment. The measured performance of the network was still lower than that of the original network. The new network was able to keep track in 6 out of 10 test sequences.

Depending on the parameters used the Particle Filter utilizing 100 particle achieved nearly online performance of about 7-9 frames per second (including image processing of full-frame PAL input).

## VI. Conclusion

We presented a stochastic model for the purpose of articulated hand motion tracking and gesture recognition. Important feature of the DBN model proposed in this paper is the incorporation of a Hidden Markov Model of gestures, that outputs a "prior" information on gesture structure that may be used to improve the results of hand tracking in an image sequence. The experiments have shown that when our model is coupled with an effective inference algorithm based on the Rao-Blackwellized Particle Filter it can be used for articulated hand tracking even in difficult, cluttered conditions. The tracker that uses knowledge on temporal gesture structure proves to be superior over the tracker with no such knowledge due to better prediction of hand motion.

## Acknowledgment

The support by the research grant N N516 070237 from Polish Ministry of Science and Higher Education is gratefully acknowledged.

## References

[1] A. D. Wilson and A. F. Bobick, "Recognition and interpretation of parametric gesture," in *Proceedings of the Sixth International Conference on Computer Vision, ICCV'98*. Washington, DC, USA: IEEE Computer Society, 1998, pp. 329–336. [Online]. Available: http://portal.acm.org/citation.cfm?id=938978.939094

[2] C.-L. Huang and S.-H. Jeng, "A model-based hand gesture recognition system," *Mach. Vision Appl.*, vol. 12, pp. 243–258, 2001. [Online]. Available: http://dx.doi.org/10.1007/s001380050144

[3] M.-H. Yang, N. Ahuja, and M. Tabb, "Extraction of 2D motion trajectories and its application to hand gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, pp. 1061–1074, 2002. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.139.6572

[4] F.-S. Chen, C.-M. Fu, and C.-L. Huang, "Hand gesture recognition using a real-time tracking method and hidden markov models," *Image Vision Comput.*, vol. 21, no. 8, pp. 745 – 758, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/B6V09-48NC790-1/2/8f12375d2a82de6de563284fd02d3f23

[5] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–287, 1989.

[6] J. Schlenzig, E. Hunter, and R. Jain, "Recursive identification of gesture inputs using hidden markov models," in *Proceedings of the Second IEEE Workshop on Applications of Computer Vision*, Dec. 1994, pp. 187–194.

[7] K. P. Murphy, "Switching kalman filters," Compaq Cambridge Research Lab, Tech. Rep. 98-10, 1998.

[8] ——, "Dynamic bayesian networks: Representation, inference and learning," Ph.D. dissertation, University of California, 2002.

[9] V. Pavlovic, "Dynamic bayesian networks for information fusion with application to human-computer interfaces," Ph.D. dissertation, University of Illinois at Urbana-Champaign, January 1999. [Online]. Available: http://www.cs.rutgers.edu/~vladimir/pub/phd.pdf

[10] B. Burger, G. Infantes, I. Ferran, and F. Lerasle, "Dbn versus hmm for gesture recognition in human-robot interaction," in *9th International workshop on Electronics, Control, Modelling, Measurement and Signals*, Mondragon, Spain, July 2009, ISBN 978-84-608-0941-8.

[11] H. Suk, B. Sin, and S. Lee, "Recognizing hand gestures using dynamic bayesian network," in *8th IEEE International Conference on Automatic Face & Gesture Recognition, FG '08*, 2008, pp. 1–6.

[12] H. Fei, "A hybrid hmm/particle filter framework for non-rigid hand motion recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, vol. 5, 2004, pp. V 889–892.

[13] M. Isard and A. Blake, "A mixed-state condensation tracker with automatic model-switching," in *Proceedings of the Sixth International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 1998, pp. 107–112. [Online]. Available: http://dl.acm.org/citation.cfm?id=938978.939138

[14] ——, "Icondensation: Unifying low-level and high-level tracking in a stochastic framework," in *Proceedings of the 5th European Conference on Computer Vision (ECCV'98), Volume I*. London, UK: Springer-Verlag, 1998, pp. 893–908. [Online]. Available: http://dl.acm.org/citation.cfm?id=645311.649045

[15] B. D. R. Stenger, "Model-based hand tracking using a hierarchical bayesian filter," Ph.D. dissertation, University of Cambridge, 2004.

[16] A. Blake and M. Isard, *Active Contours: The Application of techniques from graphics, vision, control theory and statistics to visual tracking of shapes in motion*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1998.

[17] A. Wilkowski and W. Kasprzak, "Constrained contour matching in hand posture recognition," *Image Process. Commun.*, vol. 14, no. 2-3, pp. 31–41, 2009, UTP Bydgoszcz.

[18] M. S. Arulampalam, S. Maskell, and N. Gordon, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, pp. 174–188, 2002.

[19] S. Young, "Hmms and related speech recognition technologies," in *Springer Handbook of Speech Processing*, J. Benesty, M. M. Sondhi, and Y. Huang, Eds. Berlin Heidelberg: Springer-Verlag, 2008, pp. 539–557.