Hierarchical Motion Fields

C. Barata[(a)*] M. A. T. Figueiredo[(b)] J. S. Marques[(a)]
[(a)]ISR, Instituto Superior Técnico, Lisboa, Portugal
[(b)] IT, Instituto Superior Técnico, Lisboa, Portugal

January 2019

This report describes a strategy to identify and group together motion patterns associated with different classes of agents *e.g., bikers, pedestrians, cars* in far-field surveillance scenarios.

1 Hierarchical Switched Motion Model

Let us assume that the various agents in a scene (*e.g.*, skaters, pedestrians) exhibit a finite number of motion patterns, which are specific of their class $c \in \{1, ..., C\}$. Each agent will be associated with a trajectory $x = (x_1, x_2, ..., x_L)$, where L is the length of the trajectory and $x_t \in [0,1]^2$ is the position at time instant t. The motion patterns that characterize the trajectories may be summarized into a set of K^c motion fields, where $T_k^c : [0,1]^2 \to \mathbb{R}^2$ is the $k-th$ motion field belonging to class c. Thus, we generate the position $x_t^{c_t}$ as follows

$$x_t = x_{t-1} + T_{k_t}^{c_t}(x_{t-1}) + w_{k_t}, \tag{1}$$

where $T_{k_t}^{c_t}$ is the active motion field, conditioned on class c_t, and $w_{k_t} \sim N(0, \Sigma_{k_t}^c(x_{t-1}))$ is the class-specific space-varying white noise perturbation, associated with the uncertainty of the position.

Only one motion field may be active at each time instant. However, we assume that it is possible to switch between motion fields of the same class at specific positions. Additionally, we postulate that is also possible for an agent to change classes, although with a lower probability, at certain positions of the space (*e.g.*, a car parks, a driver comes out and starts walking, becoming a pedestrian). These transitions are modeled as a hierarchical hidden Markov model (HHMM), as explained in the following section.

2 Hierarchical Motion Model

HHMM have been introduced by Fine et al. [1] as an extension of the standard HMM to problems that exhibit a hierarchical structure. The main idea of this

*This work was supported by the FCT project [PTDC/EEIPRO/0426/2014] and supported by FCT plurianual funding [UID/EEA/50009/2019].

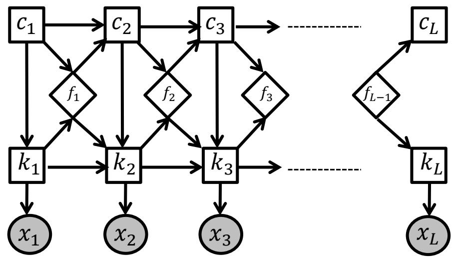

Figure 1: Graphical representation of the proposed switched model.

model is that the hidden states are organized in hierarchical levels, such that the hidden states at the uppers levels, called "internal" states, are responsible for activating states at the lower levels. Each internal state is only able to activate some of the states of the level below it, and these lower states are not shared. The process of activation is carried out until a state at the lowest level is reached. This level, usually called "production", is responsible for generating the observations, similarly to a traditional HMM [2].

Our model can be defined as a two-level HHMM (see Fig.1), where the upper level models the class and the lower level the active motion field. A binary variable f_t is used to identify the end of the production level; this allows the model to decide whether to stay in the production level ($f_t = 0$), or to leave the production level and return to the control to the upper level ($f_t = 1$). This makes it possible for the model to generate the next position by either: i) maintaining the same class; or ii) changing class.

Based on the aforementioned formulation, we define the following probabilities for our motion model [3]:

$$p(k_t = j | k_{t-1} = l, f_{t-1}, c_t = u, x_{t-1}) = \begin{cases} \tilde{B}^u_{lj}(x_{t-1}), & \text{if } f_{t-1} = 0 \\ \pi^u_j(x_{t-1}), & \text{if } f_{t-1} = 1 \end{cases}, \tag{2}$$

where $\tilde{B}^u_{lj}(x)$ is the element (l, j) of the stochastic matrix $B^u(x)$ associated with class u, and $\pi^u_j(x)$ is the initial distribution of motion field j, given the class u. Both variables are evaluated at position x. Similarly to the traditional HMM, $B^u(x)$ comprises the probabilities of transition between states l and j. Additionally, this matrix comprises the ending probabilities, *i.e.*, the probability of transition to $f_t = 1$, which we will loosely refer to *end*

$$p(f_t = 1 | k_t = j, c_t = u, x_{t-1}) = B^u_{jend}(x_{t-1}). \tag{3}$$

Thus, we also consider $B^u_{lj} = (1 - b^u_{lend}(x_{t-1}))\tilde{B}^u_{lj}(x_{t-1})$ as a rescaled version of $\tilde{B}^u_{lj}(x_{t-1})$. At the upper level, the transition between classes is also governed by a stochastic matrix $A(x)$ evaluated at position x, such that

$$p(c_t = u | c_{t-1} = v, f_{t-1}, x_{t-1}) = \begin{cases} \delta(v, u), & \text{if } f = 0 \\ A_{vu}(x_{t-1}), & \text{if } f = 1 \end{cases}. \tag{4}$$

Here $\delta(v, u)$ is the Kronecker delta and $A_{vu}(x)$ is the (v, u) element of $A(x)$ Based on this formulation, the joint probability $p(x, k, f, c)$ of a trajectory x

associated with a sequence of motion fields k, classes c, and binary variables f, is defined a follows:

$$\begin{aligned} p(x,k,f,c) = p(x_1,k_1,f_1,c_1) \prod_{t=2}^{L} p(x_t,k_t,f_t,c_t|x_{t-1},k_{t-1},f_{t-1},c_{t-1}) \\ = p(x_1,k_1,f_1,c_1) \prod_{t=2}^{L} p(x_t|x_{t-1},k_t,c_t) p(c_t|x_{t-1},c_{t-1},f_{t-1}) \\ .p(k_t|x_{t-1},c_t,f_{t-1},k_{t-1}) p(f_t|k_t,c_t,x_{t-1}). \end{aligned} \tag{5}$$

$$\begin{aligned} p(x,k,f,c) = p(x_1,k_1,f_1,c_1) \prod_{t=2}^{L} p(x_t,k_t,f_t,c_t|x_{t-1},k_{t-1},f_{t-1},c_{t-1}) \\ = p(x_1,k_1,f_1,c_1) \prod_{t=2}^{L} p(x_t|x_{t-1},k_t,c_t) p(c_t|x_{t-1},c_{t-1},f_{t-1}) \\ .p(k_t|x_{t-1},c_t,f_{t-1},k_{t-1}) p(f_t|k_t,c_t,x_{t-1}), \end{aligned} \tag{6}$$

where $p(x_t|x_{t-1},k_t,c_t)$ is a multivariate Gaussian centered in $x_{t-1}+T_{k_t}^{c_t}(x_{t-1})$ and covariance $\Sigma_{k_t}^{c}(x_{t-1})$.

2.1 Model Estimation

All of the model parameters $\theta = (\mathcal{T},\mathcal{B},\mathcal{A},\Pi,\mathbf{\Sigma})$ are defined using a regular grid of $\sqrt{n} \times \sqrt{n}$ nodes, where $\mathcal{T}$ is a dictionary of motion fields, $\mathcal{B}$ and $\mathcal{A}$ are dictionaries of fields and classes transition matrices, Π is the dictionary of motion fields probabilities, and $\mathbf{\Sigma}$ is a dictionary of covariance matrices. The parameters are estimated at the grid nodes and computed elsewhere using bilinear interpolation [4]

$$\begin{aligned} T_k^c(x) &= \sum_{i=1}^{n} T_k^{c,i} \phi^i(x) \\ B^c(x) &= \sum_{i=1}^{n} B^{c,i} \phi^i(x) \\ \mathbf{\Sigma}_k^c(x) &= \sum_{i=1}^{n} \mathbf{\Sigma}_k^{c,i} \phi^i(x) \\ A(x) &= \sum_{i=1}^{n} A^i \phi^i(x) \\ \pi^c(x) &= \sum_{i=1}^{n} \pi^{c,i} \phi^i(x), \end{aligned} \tag{7}$$

where index i identifies the parameters associated with node g^i for the $c-th$ class. The scalar $\phi^i(x)$ is the interpolation coefficient of the $i-th$ node.

The model parameters may be estimated using a set of S observed trajectories $\mathcal{X} = \left\{x^{(1)}, ..., x^{(S)}\right\}$, with variable lengths:

$$\hat{\theta} = \arg\max_{\theta} \left[\log p(\mathcal{X}|\theta) + \log p(\theta)\right]. \tag{8}$$

Since there are several hidden variables in the model (the sequences $k^{(s)}$, $c^{(s)}$, and $f^{(s)}$), we will resort to the Expectation-Maximization (EM) algorithm to alleviate the computation of the likelihood function.

$$\begin{aligned} U(\theta, \theta') &= E\left\{\log p(\mathcal{X}, \mathcal{K}|\theta)|\mathcal{X}, \theta'\right\} + \log p(\theta) \\ &= U_1(\theta, \theta') + U_2(\theta, \theta') + U_3(\theta, \theta') + U_4(\theta, \theta') + U_5(\theta, \theta'), \end{aligned} \tag{9}$$

where

$$\begin{aligned} U_1(\theta, \theta') &= \sum_{s=1}^{S} \sum_{t=2}^{L_s} \sum_{c=1}^{C} \sum_{k=1}^{K^c} \sum_{f} \gamma_{ckf}^{(s)}(t) \log \det \left(\mathbf{\Sigma}_k(x_{t-1}^{(s)})\right), \\ U_2(\theta, \theta') &= \sum_{s=1}^{S} \sum_{t=2}^{L_s} \sum_{c=1}^{C} \sum_{k=1}^{K^c} \sum_{f} \gamma_{ckf}^{(s)}(t) \|v_t^{(s)} - T_k^c(x_{t-1}^{(s)})\|^2_{\mathbf{\Sigma}_k(x_{t-1}^{(s)})}, \end{aligned}$$

$$\begin{aligned} U_3(\theta, \theta') = 2 &\sum_{s=1}^{S} \sum_{t=2}^{L_s} \sum_{c=1}^{C} \sum_{k=1}^{K^c} \xi_{ck1}^{(s)}(t) \log b_{kend}^c(x_{t-1}^{(s)}) \\ &+ \xi_{ck0}^{(s)}(t) \log(1 - b_{kend}^c(x_{t-1}^{(s)})), \end{aligned}$$

$$\begin{aligned} U_4(\theta, \theta') = 2 &\sum_{s=1}^{S} \sum_{t=2}^{L_s} \sum_{c=1}^{C} \left[\sum_{k=1}^{K^c} \eta_{ck}^{(s)}(t) \log \pi_k^c(x_{t-1}^{(s)})\right] \\ &+ \left[\sum_{p,q}^{K^c} w_{cpq}^{(s)}(t) \log \tilde{B}_{pq}^c(x_{t-1}^{(s)})\right], \\ U_5(\theta, \theta') = 2 &\sum_{s=1}^{S} \sum_{t=2}^{L_s} \sum_{j,c}^{C} \chi_{jc}^{(s)}(t) \log A_{jc}(x_{t-1}^{(s)}), \end{aligned} \tag{10}$$

where, $v_t^{(s)} = x_t^{(s)} - x_{t-1}^{(s)}$, $\gamma_{ujf}^{(s)}(t) = p(k_t^{(s)} = j, c_t^{(s)} = u, f_{t-1}^{(s)}|x^{(s)}, \theta')$ is the smooth state probability, $\xi_{ujf}^{(s)}(t) = p(k_t^{(s)} = j, c_t^{(s)} = u, f_t^{(s)}|x^{(s)}, \theta')$ gives us the ending and non-ending probabilities, $\eta_{uj}^{(s)}(t) = p(k_t^{(s)} = j, c_t^{(s)} = u, f_{t-1}^{(s)} = 1|x^{(s)}, \theta')$ is the probability of a vertical transition from the class level to the motion model one, $w_{uij}^{(s)}(t) = p(k_{t-1}^{(s)} = i, k_t^{(s)} = j, c_t^{(s)} = u, f_{t-1}^{(s)} = 0|x^{(s)}, \theta')$ is the probability of an horizontal transition at the motion model level, and $\chi_{vu}^{(s)}(t) = p(c_{t-1}^{(s)} = v, c_t^{(s)} = u, f_{t-1}^{(s)} = 1|x^{(s)}, \theta')$ is the probability of an horizontal transition at the class level. All of these values are computed in the E-step, using the generalized Baum-Welch algorithm [1].

2.2 E-step

In the E-step we will resort to the generalized Baum-Welch algorithm [1], to compute the sufficient statistics. The first step consists of estimating the forward and backward variables:

- *Forward Variables:*

$$\alpha_t^{(s)}(u,j,f_{t-1}) \triangleq p(c_t^s = u, k_t^{(s)} = j, f_{t-1}^{(s)} | x_{2:t}^{(s)})$$
$$\propto p(x_t^{(s)} | k_t^{(s)} = j, c_t^{(s)} = u, x_{t-1}^{(s)}) \sum_v^C \sum_l^{K^v} h(v,l) \sum_{g \in \{0,1\}} \alpha_{t-1}(v,l,g) \quad (11)$$

where $h(v,l)$ is the following auxiliary function

$$h(v,l) = \\ = \begin{cases} b_{lj}^v(x_{t-1}^{(s)}) & \text{if } f_{t-1}^{(s)} = 0 \\ b_{lend}^v(x_{t-1}^{(s)}) a_{vu}(x_{t-1}^{(s)}) \pi_j^u(x_{t-1}^{(s)}) & \text{if } f_{t-1}^{(s)} = 1 \end{cases}. \quad (12)$$

The variable α_2^s is initialized as follows

$$\alpha_2^{(s)}(u,j,f_{t-1}) = \begin{cases} 0 & \text{if } f_{t-1}^{(s)} = 0 \\ \pi_j^u(x_{t-1}^{(s)}) & \text{if } f_{t-1}^{(s)} = 1 \end{cases}. \quad (13)$$

- *Backward Variables:*

$$\beta_t^s(u,j,f_{t-1}) \triangleq p(x_{t+1:L^{(s)}}^{(s)} | c_t^{(s)} = u, k_t^{(s)} = j, f_{t-1}^{(s)})$$
$$= \sum_v^C \sum_l^{K^v} p(x_{t+1}^{(s)} | c_{t+1}^{(s)} = v, k_{t+1}^{(s)} = l, x_t^{(s)}) \big[\delta(u,v) b_{jl}^v(x_t^{(s)}) \beta_{t+1}(v,l,0)$$
$$+ b_{jend}^u(x_t^{(s)}) a_{uv}(x_t^{(s)}) \pi_l^v(x_t^{(s)}) \beta_{t+1}^{(s)}(v,l,1)\big]. \quad (14)$$

These variables are initialized as

$$\beta_{L^s}^s(u,j,f_{t-1}) = 1. \quad (15)$$

The second step is the calculation of the sufficient statistics, which amounts to computing the following probabilities at each time step, given the whole sequence.

- *Smooth state probability:*

$$\gamma_t^{(s)}(u,j,f_{t-1}) = p(c_t^{(s)} = u, k_t^{(s)} = j, f_{t-1}^{(s)} | x_{2:L^{(s)}}^{(s)})$$
$$\propto \alpha_t^{(s)}(u,j,f_{t-1}) \beta_t^{(s)}(u,j,f_{t-1}) \quad (16)$$

- *Horizontal transition probability at the "production"/motion fields level:*

$$\begin{aligned} w_{ulj}^{(s)}(t) &= p(k_{t-1}^{(s)} = l, k_t^{(s)} = j, c_t^{(s)} = u, f_{t-1}^{(s)} = 0|x^{(s)}) \\ &\propto \sum_{v}^{C} \delta(u,v) \sum_{g\in\{0,1\}} \alpha_{t-1}^{(s)}(v,l,g) b_{lj}^{v}(x_{t-1}^{(s)}) \\ &\times p(x_t^{(s)}|c_t^{(s)} = v, k_t^{(s)} = j, x_{t-1}^{(s)}) \beta_t^{(s)}(v,j,0) \end{aligned} \tag{17}$$

- *Not-end and End Transition probabilities:*

$$\begin{aligned} \xi_{uj0}^{(s)}(t) &= p(k_t^{(s)} = j, c_t^{(s)} = u, f_t^{(s)} = 0|x^{(s)}) \\ &\propto \sum_{v}^{C} \delta(u,v) \sum_{l}^{K^v} \sum_{g\in\{0,1\}} \alpha_t^{(s)}(v,j,g) b_{jl}^{v}(x_t^{(s)}) \\ &\times p(x_{t+1}^{(s)}|c_{t+1}^{(s)} = v, k_{t+1}^{(s)} = l, x_t^{(s)}) \beta_{t+1}^{(s)}(v,l,0) \end{aligned} \tag{18}$$

$$\begin{aligned} \xi_{uj1}^{(s)}(t) &= p(k_t^{(s)} = j, c_t^{(s)} = u, f_t^{(s)} = 1|x^{(s)}) \\ &\propto \sum_{v}^{C} \sum_{l}^{K^v} \sum_{g\in\{0,1\}} \alpha_t^{(s)}(v,j,g) b_{jend}^{v}(x_t^{(s)}) a_{uv}(x_t^{(s)}) \\ &\times p(x_{t+1}^{(s)}|c_{t+1}^{(s)} = v, k_{t+1}^{(s)} = l, x_t^{(s)}) \beta_{t+1}^{(s)}(v,l,1) \end{aligned} \tag{19}$$

- *Vertical transition probabilities:*

$$\begin{aligned} \eta_{uj}^{(s)}(t) &= p(k_t^{(s)} = j, c_t^{(s)} = u, f_{t-1}^{(s)} = 1|x^{(s)}) \\ &\propto \alpha_t^{(s)}(u,j,1) \beta_t^{(s)}(u,j,1) \end{aligned} \tag{20}$$

- *Horizontal transition probability at the "internal"/class level:*

$$\begin{aligned} \chi_{vu}^{(s)}(t) &= p(c_{t-1}^{(s)} = v, c_t^{(s)} = u, f_{t-1}^{(s)} = 1|x^{(s)}) \\ &\propto \sum_{l}^{K^v} \sum_{j}^{K^u} \sum_{g\in\{0,1\}} \alpha_{t-1} x^{(s)}(l,v,g) b_{lend}^{v}(x_{t-2}^{(s)}) a_{vu}(x_{t-1}^{(s)}) \\ &\times \pi_j^u(x_{t-1}^{(s)}) p(x_t^{(s)}|c_t^{(s)} = u, k_t^{(s)} = j, x_{t-1}^{(s)}) \beta_{t+1}^{(s)}(u,j,1) \end{aligned} \tag{21}$$

2.3 M-step

The M-step consists of optimizing (9) w.r.t to θ, which is accomplished by taking the derivatives.

- *Motion Fields:*

$$\frac{\partial U}{\partial \mathcal{T}_\kappa^\zeta} = \frac{\partial U_2}{\partial \mathcal{T}_\kappa^\zeta} + \frac{\log p(\theta)}{\partial \mathcal{T}_\kappa^\zeta} = 0, \tag{22}$$

where we define $\log p(\theta) = \alpha \|\Delta \mathcal{T}_\kappa^\zeta\|_2^2 + \beta \|\mathcal{T}_\kappa^\zeta\|_p^p$ to ensure both smoothness and sparsity of the fields.

The aforementioned may be expressed as a convex optimization problem, such that

$$\min_{\mathcal{T}_\kappa} \|W_\kappa \mathrm{vec}(V - \mathcal{T}_\kappa^\zeta \Phi)\|_2^2 + \alpha \|\Delta \mathcal{T}_\kappa^\zeta\|_F + \beta \|\mathcal{T}_\kappa^\zeta\|_p^p, \tag{23}$$

where $V \in \mathbb{R}^{2\times L}$ ($L = \sum_s L_s - S$) comprises the agents' velocities ($v = x_t - x_{t-1}$) off all trajectories, $\Phi \in \mathbb{R}^{n \times L}$ is the matrix of the interpolation coefficients for all the trajectories, and W_κ is a block diagonal matrix of the form

$$W_k = \begin{bmatrix} \sqrt{\gamma_k^{(1)}(2)}\mathbf{\Sigma_k}^{-\frac{1}{2}} & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & \dots & \vdots & \dots & \vdots & \dots & \vdots \\ 0 & \dots & \sqrt{\gamma_k^{(1)}(L_1)}\mathbf{\Sigma_k}^{-\frac{1}{2}} & \dots & 0 & \dots & 0 \\ \vdots & \dots & \vdots & \dots & \vdots & \dots & \vdots \\ 0 & \dots & 0 & \dots & \sqrt{\gamma_k^{(S)}(2)}\mathbf{\Sigma_k}^{-\frac{1}{2}} & \dots & 0 \\ \vdots & \dots & \vdots & \dots & \vdots & \dots & \vdots \\ 0 & \dots & 0 & \dots & 0 & \dots & \sqrt{\gamma_k^{(S)}(L_S)}\mathbf{\Sigma_k}^{-\frac{1}{2}} \dots \end{bmatrix},$$

where $\gamma_{\kappa_t}^{(s)} = \sum_c \sum_f \gamma_t^{(s)}(c, k, f)$.

The objective function to be minimized is a sum of norms. Therefore, it is convex and can be solved using a software package for convex problems, such as CVX [**?**].

- *Noise covariance matrices:*

$$\frac{\partial U}{\partial \mathbf{\Sigma}_\kappa^{\lambda,\zeta}} = \frac{\partial U_1}{\partial \mathbf{\Sigma}_\kappa^{\lambda,\zeta}} + \frac{\partial U_2}{\partial \mathbf{\Sigma}_\kappa^{\lambda,\zeta}} = 0, \tag{24}$$

The derivatives $\frac{\partial U}{\partial \mathbf{\Sigma}_\kappa^{\lambda,\zeta}}$ are equal to

$$\begin{aligned} \frac{\partial}{\partial \mathbf{\Sigma}_\kappa^{\lambda,\zeta}} E[\log p(\mathcal{X}, \mathcal{K}|\theta)|\mathcal{X}, \hat{\theta}] \quad = \quad & \sum_{s=1}^{S} \sum_{t=1}^{L_s} \gamma_\kappa^s(t) \phi_\lambda(x_{t-1}^{(s)}) [\left(\sum_{n=1}^{N} \mathbf{\Sigma}_\kappa^{n,\zeta} \phi^n(x_{t-1}^{(s)}) \right)^{-1} - \\ - \quad & \left(\left(\sum_{n=1}^{N} \mathbf{\Sigma}_\kappa^{n,\zeta} \phi^n(x_{t-1}^{(s)}) \right)^{-1} \mathbf{G}_t^{(s)} \left(\sum_{n=1}^{N} \mathbf{\Sigma}_\kappa^{n,\zeta} \phi^n(x_{t-1}^{(s)}) \right)^{-1} \right)^T, \end{aligned} \tag{25}$$

where

$$\mathbf{G}_t^{(s)} = (v_t^{(s)} - \mathcal{T}_\kappa \Phi(x_{t-1}^{(s)}))(v_t^{(s)} - \mathcal{T}_\kappa \Phi(x_{t-1}^{(s)}))^T. \tag{26}$$

Setting $\frac{\partial U(\theta,\hat{\theta})}{\partial \mathbf{\Sigma}_\alpha^\gamma} = 0$ does not have an explicit solution. Thus, we resort to the gradient descent method followed by a projection on the set of semidefinite positive matrices.

- *Class transition matrices:*

$$\frac{\partial U}{\partial A^\lambda_{\zeta\rho}} = \frac{\partial U_5}{\partial A^\lambda_{\zeta\rho}} = 0, \tag{27}$$

The derivatives are computed as follows

$$\begin{aligned}
\frac{\partial U_5}{\partial A^\lambda_{\zeta\rho}} &= \frac{\partial}{\partial A^\lambda_{\zeta\rho}} \sum_{s=1}^{S} \sum_{t=2}^{L_s} \sum_{u,v}^{C} \chi^{(s)}_{uv}(t) \log \sum_{i=1}^{n} A^i_{uv} \phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S} \sum_{t=2}^{L_s} \frac{\partial}{\partial A^\lambda_{\zeta\rho}} \sum_{u,v}^{C} \chi^{(s)}_{uv}(t) \log \sum_{i=1}^{n} A^i_{uv} \phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S} \sum_{t=2}^{L_s} \chi^{(s)}_{\zeta\rho}(t) \frac{\partial}{\partial A^\lambda_{\zeta\rho}} \log \sum_{i=1}^{n} A^i_{\zeta\rho} \phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S} \sum_{t=2}^{L_s} \frac{\chi^{(s)}_{\zeta\rho}(t) \phi^\lambda(x^{(s)}_{t-1})}{\sum_{i=1}^{n} A^i_{\zeta\rho} \phi^i(x^{(s)}_{t-1})}
\end{aligned} \tag{28}$$

Setting $\frac{\partial U_5}{\partial A^\lambda_{\zeta\rho}} = 0$ does not have an explicit solution. Thus, we resort to the gradient descent method followed by a projection on the simplex.

- *Class-specific fields transition matrices:*

$$\begin{aligned}
\frac{\partial U_4}{\partial \tilde{B}^{\lambda\zeta}_{\kappa\rho}} &= \frac{\partial}{\tilde{B}^{\lambda\zeta}_{\kappa\rho}} \sum_{s=1}^{S} \sum_{t=2}^{L_s} \sum_{c=1}^{C} \sum_{p,q}^{K^c} w^{(s)}_{cpq}(t) \log \sum_{i=1}^{n} \tilde{B}^{ic}_{pq} \phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S} \sum_{t=2}^{L_s} \frac{\partial}{\tilde{B}^{\lambda\zeta}_{\kappa\rho}} \sum_{c=1}^{C} \sum_{p,q}^{K^c} w^{(s)}_{cpq}(t) \log \sum_{i=1}^{n} \tilde{B}^{ic}_{pq} \phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S} \sum_{t=2}^{L_s} w^{(s)}_{\zeta\kappa\rho}(t) \frac{\partial}{\tilde{B}^{\lambda\zeta}_{\kappa\rho}} \log \sum_{i=1}^{n} \tilde{B}^{ic}_{pq} \phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S} \sum_{t=2}^{L_s} \frac{w^{(s)}_{\zeta\kappa\rho}(t) \phi^\lambda(x^{(s)}_{t-1})}{\sum_{i=1}^{n} \tilde{B}^{i\lambda}_{\kappa\rho} \phi^i(x^{(s)}_{t-1})}
\end{aligned} \tag{29}$$

Setting $\frac{\partial U_4}{\partial \tilde{B}^{\lambda\zeta}_{\kappa\rho}} = 0$ does not have an explicit solution. Thus, we resort to the gradient descent method followed by a projection on the simplex.

- *Class-specific fields' probabilities (vertical transitions):*

$$\begin{aligned}
\frac{\partial U_4}{\partial \pi_\kappa^{\lambda\zeta}} &= \frac{\partial}{\partial \pi_\kappa^{\lambda\zeta}} \sum_{s=1}^{S}\sum_{t=2}^{L_s}\sum_{c=1}^{C}\sum_{k}^{K^c} \eta_{ck}^{(s)}(t) \log \sum_{i=1}^{n} \pi_k^{ic} \phi^i(x_{t-1}^{(s)}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \frac{\partial}{\partial \pi_\kappa^{\lambda\zeta}} \sum_{c=1}^{C}\sum_{k}^{K^c} \eta_{ck}^{(s)}(t) \log \sum_{i=1}^{n} \pi_k^{ic} \phi^i(x_{t-1}^{(s)}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \eta_{\zeta\kappa}^{(s)}(t) \frac{\partial}{\partial \pi_\kappa^{\lambda\zeta}} \log \sum_{i=1}^{n} \pi_k^{ic} \phi^i(x_{t-1}^{(s)}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \frac{\eta_{\zeta\kappa}^{(s)}(t) \phi^\lambda(x_{t-1}^{(s)})}{\sum_{i=1}^{n} \pi_\kappa^{\lambda\zeta} \phi^i(x_{t-1}^{(s)})} \qquad (30)
\end{aligned}$$

Setting $\frac{\partial U_4}{\partial \pi_\kappa^{\lambda\zeta}} = 0$ does not have an explicit solution. Thus, we resort to the gradient descent method followed by a projection on the simplex.

- *Class-specific ending and not ending probabilities:*

We will start by defining

$$B_{j\backslash end}^{u} = 1 - B_{jend}^{u}. \qquad (31)$$

Thus, $U_3(\theta, \theta')$ becomes

$$\begin{aligned}
U_3(\theta, \theta') = 2 \sum_{s=1}^{S}\sum_{t=2}^{L_s}\sum_{c=1}^{C}\sum_{k=1}^{K^c} \xi_{ck1}^{(s)}(t) \log B_{kend}^{c}(x_{t-1}^{(s)}) \\
+ \xi_{ck0}^{(s)}(t) \log B_{k\backslash end}^{c}(x_{t-1}^{(s)}), \qquad (32)
\end{aligned}$$

Now, it is possible to compute the derivatives

$$\begin{aligned}
\frac{\partial U_3}{\partial B_{\kappa end}^{\lambda\zeta}} &= \frac{\partial}{\partial B_{\kappa end}^{\lambda\zeta}} \sum_{s=1}^{S}\sum_{t=2}^{L_s}\sum_{c=1}^{C}\sum_{k=1}^{K^c} \xi_{ck1}^{(s)}(t) \log \sum_{i=1}^{n} B_{kend}^{ic} \phi^i(x_{t-1}^{(s)}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \frac{\partial}{\partial B_{\kappa end}^{\lambda\zeta}} \sum_{c=1}^{C}\sum_{k=1}^{K^c} \xi_{ck1}^{(s)}(t) \log \sum_{i=1}^{n} B_{kend}^{ic} \phi^i(x_{t-1}^{(s)}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \xi_{\zeta\kappa 1}^{(s)}(t) \frac{\partial}{\partial B_{\kappa end}^{\lambda\zeta}} \log \sum_{i=1}^{n} B_{kend}^{ic} \phi^i(x_{t-1}^{(s)}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \frac{\xi_{\zeta\kappa 1}^{(s)}(t) \phi^\lambda(x_{t-1})^{(s)}}{\sum_{i=1}^{n} B_{\kappa end}^{i\zeta} \phi^i(x_{t-1}^{(s)})} \qquad (33)
\end{aligned}$$

$$
\begin{aligned}
\frac{\partial U_3}{\partial B^{\lambda\zeta}_{\kappa\setminus end}} &= \frac{\partial}{\partial B^{\lambda\zeta}_{\kappa\setminus end}} \sum_{s=1}^{S}\sum_{t=2}^{L_s}\sum_{c=1}^{C}\sum_{k=1}^{K^c} \xi^{(s)}_{ck0}(t) \log \sum_{i=1}^{n} B^{ic}_{k\setminus end}\phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \frac{\partial}{\partial B^{\lambda\zeta}_{\kappa\setminus end}} \sum_{c=1}^{C}\sum_{k=1}^{K^c} \xi^{(s)}_{ck0}(t) \log \sum_{i=1}^{n} B^{ic}_{k\setminus end}\phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \xi^{(s)}_{\zeta\kappa 1}(t) \frac{\partial}{\partial B^{\lambda\zeta}_{\kappa\setminus end}} \log \sum_{i=1}^{n} B^{ic}_{k\setminus end}\phi^i(x^{(s)}_{t-1}) \\
&= \sum_{s=1}^{S}\sum_{t=2}^{L_s} \frac{\xi^{(s)}_{\zeta\kappa 1}(t)\phi^{\lambda}(x_{t-1})^{(s)}}{\sum_{i=1}^{n} B^{i\zeta}_{\kappa\setminus end}\phi^i(x^{(s)}_{t-1})} \qquad (34)
\end{aligned}
$$

Similarly to the previous parameters, $B_{\kappa end}$ and $B_{\kappa\setminus end}$ are also estimated using the gradient descent method, followed by a projection on the simplex.

References

[1] S. Fine, Y. Singer, and N. Tishby, "The hierarchical hidden Markov model: Analysis and Applications," *Machine learning*, vol. 32, no. 1, pp. 41–62, 1998.

[2] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 257286, 1989.

[3] W. Kong, Z. Y. Dong, D. J. Hill, J. Ma, J. H. Zhao, and F. J. Luo, "A hierarchical hidden Markov model framework for home appliance modeling," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3079–3090, 2018.

[4] R. Szeliski, *Computer vision: algorithms and applications*, 2010.