Approximate Inference for Generic Likelihoods via Density-Preserving GMM Simplification

Lei Yu Tianyu Yang Antoni B. Chan
Dept. of Computer Science, City University of Hong Kong

Contribution

- We propose a Density-Preserving Hierarchical EM (DPHEM) algorithm to reduce a Gaussian Mixture Model (GMM) by maximizing a variational lower bound of the expected log-likelihood of a set of virtual samples.
- We propose an efficient algorithm for approximating an arbitrary likelihood function as a sum of scaled Gaussian (SSG).
- We apply an unified recursive Bayesian filtering framework with arbitrary likelihood to visual tracking, where the posterior is represented as a GMM.

Density-Preserving Hierarchical EM Algorithm

- **Goal**
  - Reduce the number of components in a GMM $p(y|\Theta^{(b)}) = \sum_{i=1}^{K_b} \pi_i^{(b)} p(y|\theta_i^{(b)})$ to $p(y|\Theta^{(r)}) = \sum_{j=1}^{K_r} \pi_j^{(r)} p(y|\theta_j^{(r)})$ with $K_r \ll K_b$.
- **Principle**
  - Define a set of i.i.d. virtual samples $Y = \{y_1, y_2, \cdots, y_N\}$ with each $y_n \sim \Theta^{(b)}$.
  - The reduced model $\Theta^{(r)}$ is obtained by maximizing the *expected* log-likelihood of the reduced model $\Theta^{(r)}$ with respect to the virtual samples,

$$\mathcal{J}(\Theta^{(r)}) = \mathbb{E}_{Y|\Theta^{(b)}}[\log p(Y|\Theta^{(r)})] = \sum_i \pi_i^{(b)} \mathbb{E}_{Y|\theta_i^{(b)}}[\log p(Y|\Theta^{(r)})].$$

- **Variational Lower Bound**

$$\mathcal{J}_{DP}(\Theta^{(r)}) = \max_{z_{ij}} \sum_i \sum_j \pi_i^{(b)} z_{ij} \left\{ \log \frac{\pi_j^{(r)}}{z_{ij}} + N\mathbb{E}_{y|\theta_i^{(b)}}[\log p(y|\theta_j^{(r)})] \right\} \leq \mathcal{J}(\Theta^{(r)}).$$

- **Solution for GMMs**
  - E-Step:

$$\hat{z}_{ij} = \frac{\pi_j^{(r)} \exp(N\mathbb{E}_{y|\theta_i^{(b)}}[\log p(y|\theta_j^{(r)})])}{\sum_{j'=1}^{K_r} \pi_{j'}^{(r)} \exp(N\mathbb{E}_{y|\theta_i^{(b)}}[\log p(y|\theta_{j'}^{(r)})])},$$

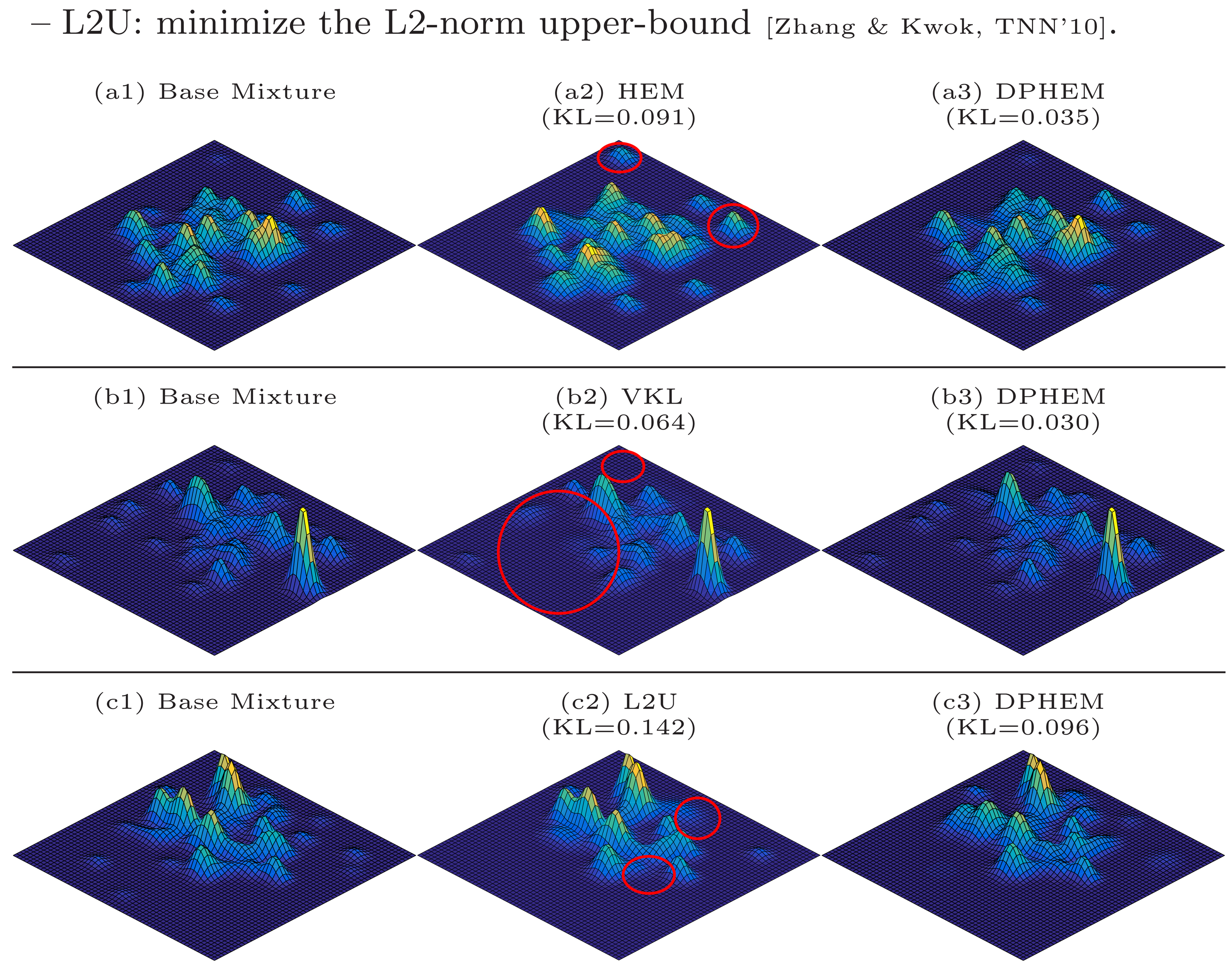
$$\mathbb{E}_{y|\theta_i^{(b)}}[\log p(y|\theta_j^{(r)})] = \log \mathcal{N}(\mu_i^{(b)}|\mu_j^{(r)}, \Sigma_j^{(r)}) - \frac{1}{2}\mathrm{tr}\{(\Sigma_j^{(r)})^{-1}\Sigma_i^{(b)}\}.$$

  - M-Step:

$$\hat{\pi}_j^{(r)} = \sum_{i=1}^{K_b} \pi_i^{(b)} \hat{z}_{ij}, \qquad \hat{\mu}_j^{(r)} = \frac{1}{\hat{\pi}_j^{(r)}} \sum_{i=1}^{K_b} \hat{z}_{ij} \pi_i^{(b)} \mu_i^{(b)},$$

$$\hat{\Sigma}_j^{(r)} = \frac{1}{\hat{\pi}_j^{(r)}} \sum_{i=1}^{K_b} \hat{z}_{ij} \pi_i^{(b)} [\Sigma_i^{(b)} + (\mu_i^{(b)} - \hat{\mu}_j^{(r)})(\mu_i^{(b)} - \hat{\mu}_j^{(r)})^T].$$

- **Comparison with Other Simplifying Algorithms**
  - HEM: component clustering [Vasconcelos & Lippman, NIPS'98].
  - VKL: minimize the variational upper-bound of KLD [Brubaker, et. al, TPAMI'16].
  - L2U: minimize the L2-norm upper-bound [Zhang & Kwok, TNN'10].



Recursive Bayesian Inference

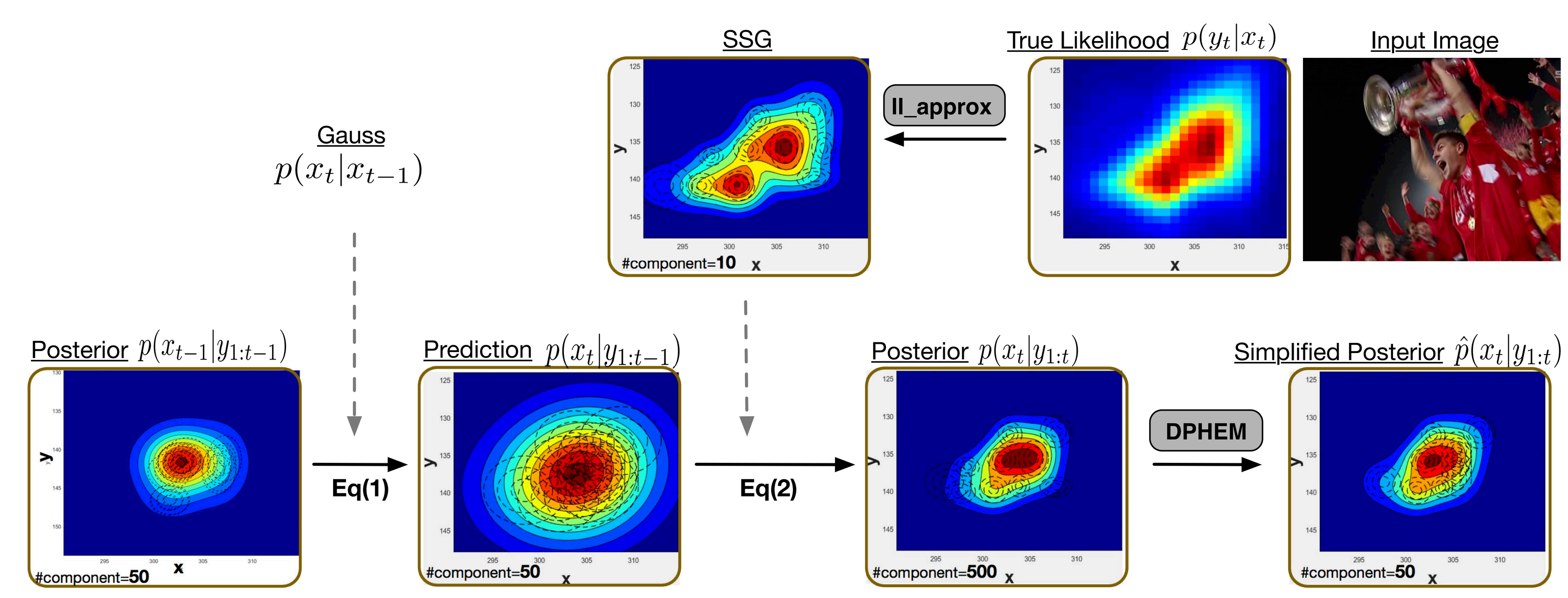
- **Goal**
  - Calculate the posterior distribution of latent state variable x_t conditioned on all observations so far $y_{1:t} = \{y_1, \cdots, y_t\}$.
- **First-order Markov Framework**
  - Predict the current state x_t using the previous posterior distribution $p(x_{t-1}|y_{1:t-1})$ and transition model $p(x_t|x_{t-1})$:

$$p(x_t|y_{1:t-1}) = \int p(x_t|x_{t-1}) p(x_{t-1}|y_{1:t-1}) dx_{t-1}. \quad (1)$$

  - Factor in the current observation y_t using the observation model $p(y_t|x_t)$:

$$p(x_t|y_{1:t}) \propto p(y_t|x_t) p(x_t|y_{1:t-1}). \quad (2)$$

  - We model the posterior $p(x_t|y_{1:t})$ as a GMM, and likelihood $p(y_t|x_t)$ as a SSG.
  - The number of components in the GMM posterior increases in each iteration, and we use DPHEM to reduce the GMM to a manageable size.
- **Framework on Visual Tracking**
  - State x_t is the target position; $p(y_t|x_t)$ is the score from the observation model.



Lower-bound Likelihood Approximation

- **Goal**
  - Approximate arbitrary likelihood function $f(x) = p(y|x)$ with a sum of scaled Gaussian (SSG) $f(x) = \sum_k f^{(k)}(x)$, where $f^{(k)}(x)$ is a scaled Gaussian.
- **Iterative Fitting of $f^{(k)}(x)$**
  - Residual set $\mathcal{D}^{(k)} = \{(x_i, r_i)\}_{i=1}^N$, where $r_i = p_i - f^{(k-1)}(x_i), \forall i$ (Initially, $r_i = p_i$).
  - Calculate log-residuals, $\ell_i = \log r_i$, and find maximum, $m = \mathrm{argmax}_i\, \ell_i$.
  - Anchor the peak of $f^{(k)}(x)$ to highest point in log-space (x_m, ℓ_m),

$$h^{(k)}(x) = -(x - x_m)^T W_k (x - x_m) + \ell_m, \quad f^{(k)}(x) = \exp(h^{(k)}(x)).$$

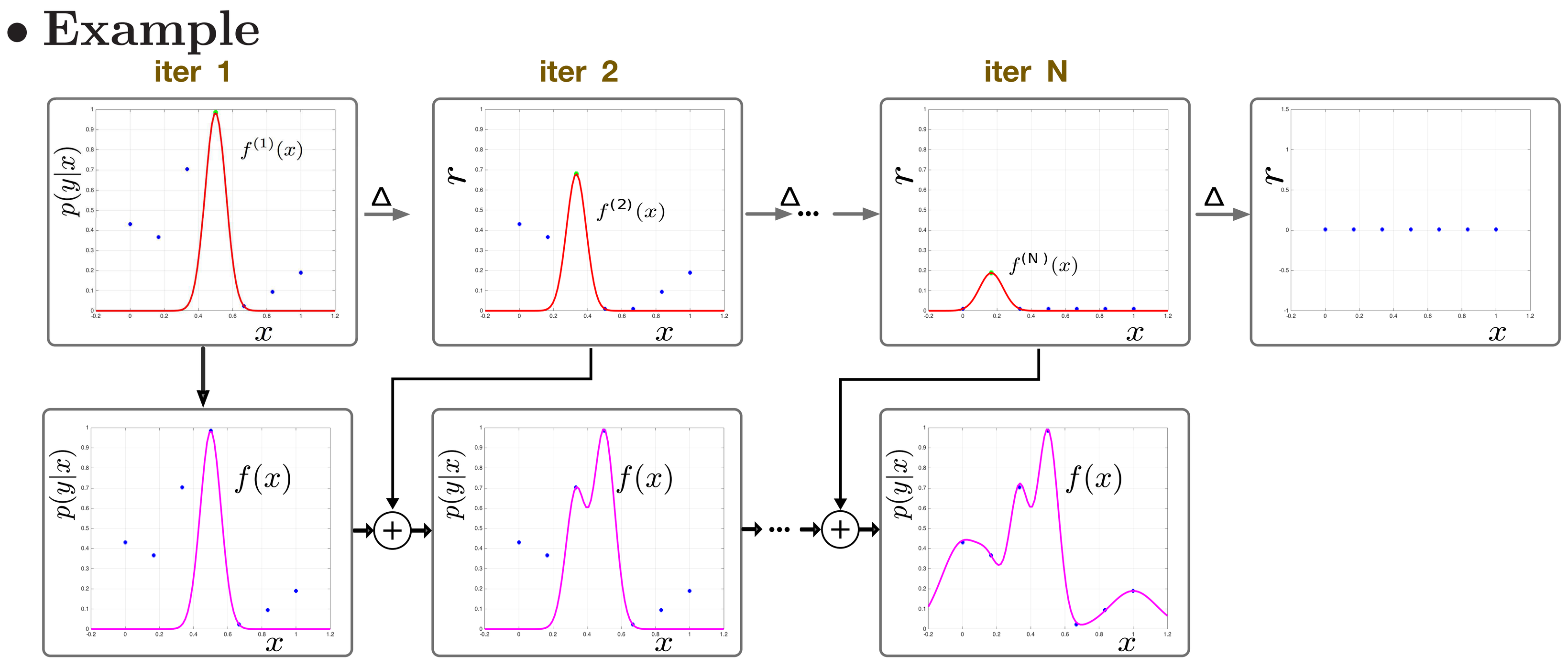
  - Find the precision matrix W_k by minimizing the squared error, while ensuring $h^{(k)}(x)$ is a lower-bound to the residuals,

$$W_k^* = \operatorname*{argmin}_{W_k} \frac{1}{2} \sum_{i=1}^N (\ell_i - h^{(k)}(x_i))^2 \quad \text{s.t.} \quad \ell_i - h^{(k)}(x_i) \geq 0, \forall i.$$

- **Diagonal Precision**
  - Assuming $W = \mathrm{diag}(w)$ results in a constrained least-squares problem:

$$w^* = \operatorname*{argmin}_w \frac{1}{2} \sum_{i=1}^N (\tilde{\ell}_i + w^T \tilde{x}_i)^2 \quad \text{s.t.} \quad \tilde{\ell}_i + w^T \tilde{x}_i \geq 0, \ \forall i, \ w \geq 0,$$

  where $\tilde{x}_i = (x_i - x_m)^2$ is the element-wise square difference, and $\tilde{\ell}_i = \ell_i - \ell_m$.
- **Example**



  - Δ indicates calculation of the residuals: $r_i = p(y_i|x_i) - f^{(k)}(x_i)$.

Experiments

- **Synthetic 2d GMM:** reduce randomly-generated GMMs with 2,500 components.
- **Visual Tracking:** tracking with recursive Bayesian inference on 50 video sequences [Y.Wu et al. CVPR'13].
- **Belief Propagation:** use GMM potentials on 4-node graph without sampling.