Probabilistic Tracking in a Metric Space

Kentaro Toyama, Andrew Blake

Presented by: Jongwoo Lim, Nov 21 2002

Contents

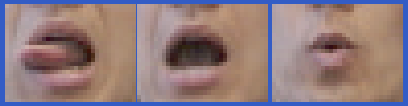
Exemplar + Transformation

Observation z as a transformed exemplar x.

$$z \simeq \mathcal{T}_\alpha x$$

Transformation $\mathcal{T}_\alpha$ is defined analytically in advance, then the set of exemplars $\mathcal{X}$ is learned from a training sequence $\mathcal{Z}^*$.

$$\mathcal{X} = \{x_k,\ k = 1, \ldots, K\}$$



Rigid Trans

$\alpha = (\mathbf{u}, \theta, s)$

$\mathcal{T}_\alpha \mathbf{r} = \mathbf{u} + sR(\theta)\mathbf{r}$

Metric Distance

What is ‘metric’?

$$\rho(a,b) \geq 0 \qquad \rho(a,b) = 0 \text{ iff } a = b$$

$$\rho(a,b) = \rho(b,a) \qquad \rho(a,b) + \rho(b,c) \geq \rho(a,c)$$

Metric space vs. Vector space

there only exist ‘distances’ among the elements.

Exemplars will be interpreted probabilistically, so that the uncertainty in approximation is accounted for explicitly.

Image z as a state vector $X = (\alpha, k)$ from $\mathcal{T}_\alpha \, x_k$.

Metric Mixture (M^2) Model

Exemplars as Mixture Centers

$$p(z|X) \;=\; p(z|\tilde{z}) \;\simeq\; \frac{1}{Z}\, e^{-\lambda\, \rho(z,\tilde{z})}$$

Metric-based Mixture Kernels

$$p(z|\alpha) \;\propto\; \sum_k \pi_k \,\frac{1}{Z}\, e^{-\lambda\, \rho(z,\mathcal{T}_\alpha x_k)}$$

when only motion is tracked (π_k : mixture weight).

$\mathbf{M}^2$ Model : Partition Function

When the distance function $\rho(z, \tilde{z})$ is approximately quadratic, we can consider the distribution as approximate Gaussian.

$$\lambda = \frac{1}{2\sigma^2}, \qquad Z \propto \sigma^d$$

From this, $\rho|\tilde{z} \equiv \rho(z, \tilde{z})$ is a $\sigma^2\chi^2_d$ random variable.

λ, σ are to be learned from training data.

Learning Algorithms

From the training sequence Z^*, we need to learn

- Mixture centers $\mathcal{X} = \{x_k, k = 1, \ldots, K\}$
- Component distributions $p(z|X)$
- Transition distribution $p(X_t|X_{t-1})$, $p(X_1)$
- Kernel parameters σ, d

Learning Mixture Kernel Centers ($\mathcal{X}$)

Align the training set

- $z_0^* \leftarrow \arg\min_{z \in \mathcal{Z}^*} \max_{z' \in \mathcal{Z}^* - \{z\}} \rho(z^*, z')$
- $\alpha_t^* =: \arg\min_{\alpha} \rho(T_\alpha^{-1} z_t^*, z_0^*), \quad x_t^* = T_{\alpha_t^*}^{-1} z_t^*$
- Initialize ceters $\tilde{x}_k$ so to be evenly spaced.

Iterate until convergence

- Cluster training data $\mathcal{C}_k = \{x_t^* : k_t(x_t^*) = k\}$ where $k_t(x_t^*) = \arg\min_k \rho(x_t^*, \tilde{x}_k)$.
- $\tilde{x}_k = \arg\min_{x \in \mathcal{C}_k} \max_{x' \in \mathcal{C}_k - \{x\}} \rho(x, x')$

Learning M^2 Kernel Parameters

For a vaildation set $\mathcal{Z}_v = \mathcal{Z} - \{\tilde{z}_k\}$, $\rho_v(z_v) = \rho(z_v, \tilde{z}_v)$ is $\sigma^2 \chi^2_d$ distributed.

d_k, σ_k can be estimated with moments $\bar{\rho}$ and $\bar{\rho^2}$.

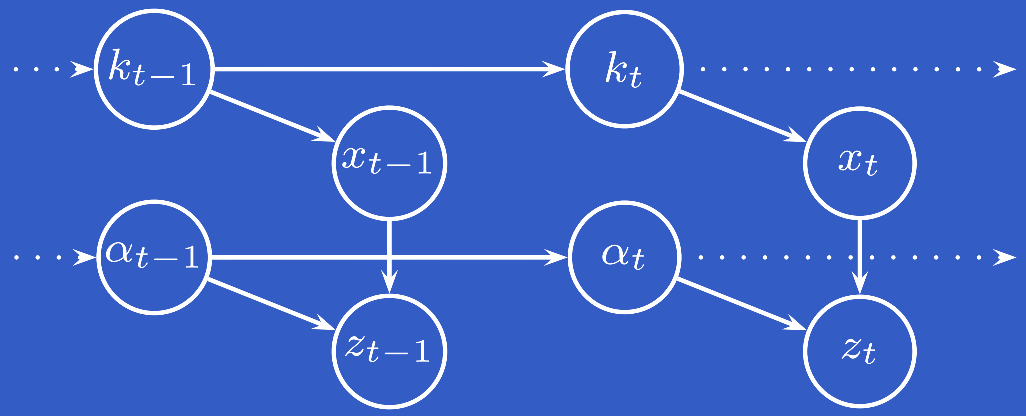
$$d_k = \frac{2\ \bar{\rho}_k^2}{\bar{\rho^2}_k - \bar{\rho}_k^2}\ , \qquad \sigma_k = \sqrt{\bar{\rho}_k / d_k}$$

$$\bar{\rho}_k = E[\,\rho_v(z_v)\,] \qquad \bar{\rho^2}_k = E[\,\rho_v(z_v)^2\,] \qquad \text{for } z_v \in \mathcal{C}_k$$

Learning Dynamics

Learn $p(X_t|X_{t-1})$

- Learn a Markov matrix M for $p(k_t|k_{t-1})$ by histogramming transitions.
- Run a first-order auto-regressive process (ARP) for $p(\alpha_t|\alpha_{t-1})$, with coefficients calculated using the Yule-Walker algorithm.



Practical Tracking

Classical forward algorithm gives

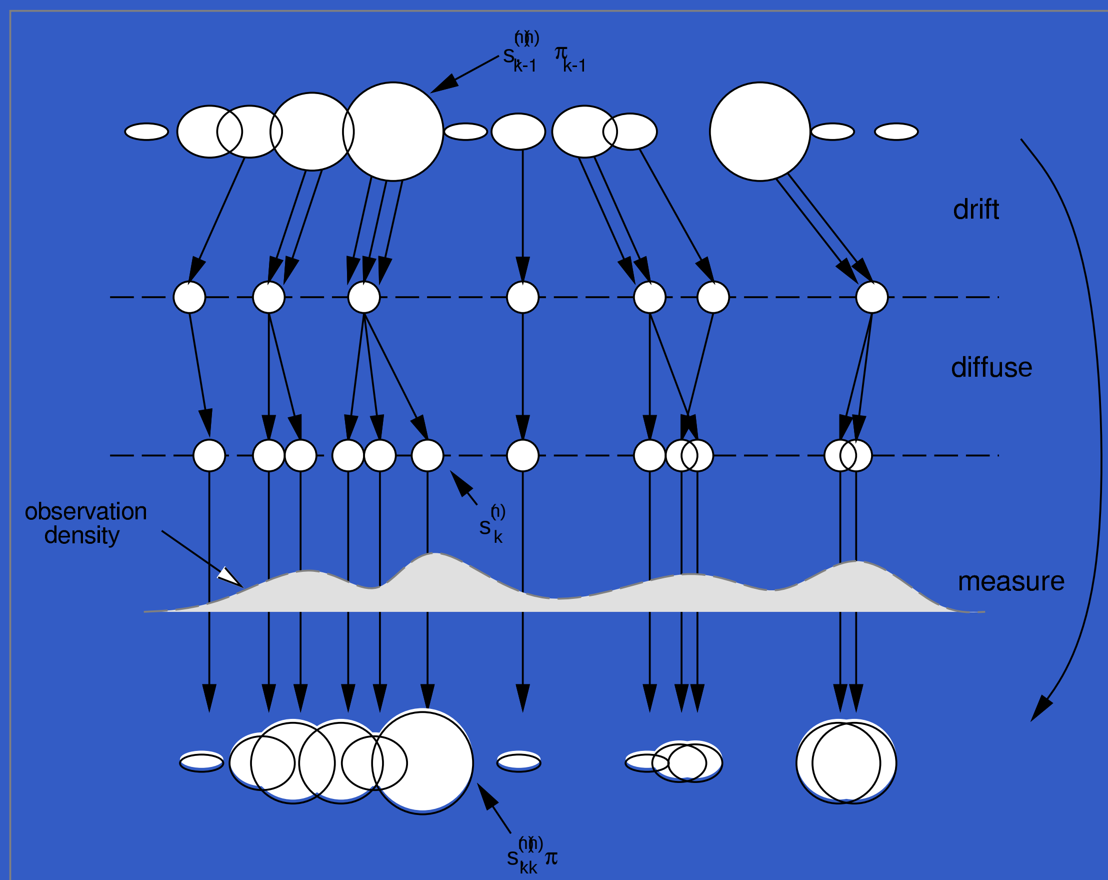
$$\begin{aligned} p_t(X_t) &\equiv p(X_t | Z_1, Z_2, \ldots, Z_t) \\ &= \sum_{k_{t-1}} \int_{\alpha_{t-1}} p(z|X_t)\, p(X_t|X_{t-1})\, p_{t-1}(X_{t-1}) \end{aligned}$$

Practically tracking is done using a particle filter.

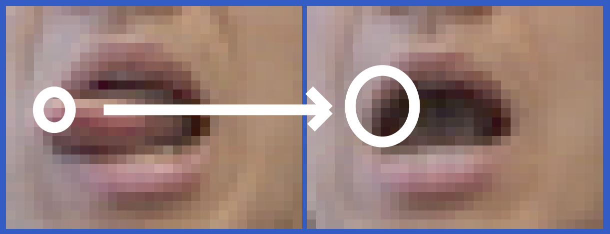
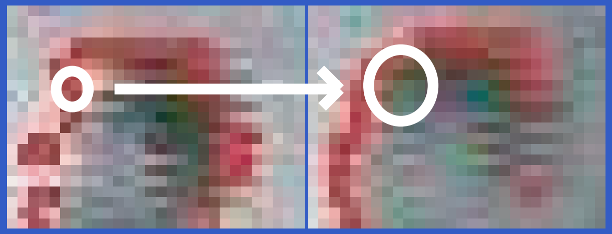
$$\hat{X}_t = \arg\max \; p_t(X_t)$$

Practical Tracking : Particle Filtering

CONDENSATION algorithm



Modelling Images and Observations

Exemplar	Patches	Curves
Description	image subregion	parametric curve
Interpretation	$I_z(\mathbf{r}) = \sum_{i=1}^{d} I_i(\mathbf{r})\, y_i$	$\mathbf{r}_z(s) = \sum_{i=1}^{d} \mathbf{r}_i(s) y_i$
Metric (ρ)	Shuffle distance 	Chamfer distance 

- Shuffle dist. - distance with the most similar pixel in its neighbor
- Chamfer dist. - distance to the nearest pixel in the binary images

Results : Dynamics

A randomly generated sequence using leaned dynamics

Results : Validity of M^2 model

Object	Avg. C Size	Actual DOF	d	σ
Synthesized ellipse	100	1	0.8	89.1
	100	2	1.2	93.3
	100	3	1.5	86.4
	100	4	3.6	71.9
Person Contour	5	?	2.8	21.6
	10	?	4.1	14.4
	20	?	5.1	18.3
	40	?	5.0	17.9

Results : Tracking

- Walking sequence
- Mouth tracking

Conclusion

- Combines exemplars in a metric space with a probabilistic framework.
- Generality
  - Ability to learn both object and noise models
  - Metrics can be chosen without significant restrictions

Acknowledgement

Thanks to

- Prof. Belongie for reviewing and commenting this slides
- Sameer for deriving the parameter estimation equations