Bayes Filter for Gesture Recognition

Variables

Hidden State Space
$\mathbb{Z} = \{z_{\varnothing}^t, z_1^t, ..., z_m^t\}$
Where there are m objects and t represents the timestamp on the state.
ME: Is there a way we should formalize how the state space changes: point cloud changes, number of objects changes, etc.
Each state z_i^t represents an object as a series of 3D points at time t.
ME: Consider combinations of objects? This would allow a good model of "The yellow objects". Initial model is for single objects only, but that is definitely a worthwhile extension.

Observations
$\mathbb{O} = \{o^1, ...o^n\}$
Where the superscript represents the timestamp for each observation.
Where $o^i = \{\vec{l^i}, \vec{r^i}, \vec{h^i}, s^i\}$
Each observation consists of four parts:
$\vec{l^i}$, the left arm vector at time i
$\vec{r^i}$, the right arm vector
$\vec{h^i}$, the head vector
s^i, speech
Since not all components of the observation tuple are guaranteed to be present at the same time, any of the four can take on a null value

Transition Function
The transition function $\mathbb{T}(z_i^t, z_k^{t+1})$ returns the probability of state z_i^t transitioning to z_k^{t+1}.
ST: I think we should use maybe a Poisson Process.
We could also weight the transition functions based on shared properties.

Let $\mathbb{N}$ represent the probability of seeing a specific sample with a normal distribution of the specified parameters, θ represent the angle between the input vector and the sample point, and u, w, x, and y be exponents for weighting each sample appropriately.

Equations

ME: What did you mean by separating out the time and measurement update. I thought We wish to know the most likely state (object being referenced) given our observations and previous state estimation, namely:

$$\underset{z_i^t}{\operatorname{argmax}}\, P(o^t|z_i^t) \propto \underset{z_i^t}{\operatorname{argmax}}[P(z_i^t|o^t) * \sum_{z_k^{t-1} \in \mathbb{Z}} P(z_i^t|z_k^{t-1})]$$

Where:

$P(z_i^t|z_k^{t-1}) = \mathbb{T}(z_k^{t-1}, z_i^t) * P(z_k^{t-1})$

$P(z_k^0) = \frac{1}{m+1}$

$P(z_i^t|o^t) = P(z_i^t|l^t)^u * P(z_i^t|r^t)^w * P(z_i^t|h^t)^x * P(z_i^t|s^t)^y$

$P(z_i^t|l^t) = \prod_{p \in z_i^t} \mathbb{N}(\mu_l = 0, \sigma_l, \theta(l^t, p))$

$P(z_i^t|r^t) = \prod_{p \in z_i^t} \mathbb{N}(\mu_r = 0, \sigma_r, \theta(r^t, p))$

$P(z_i^t|h^t) = \prod_{p \in z_i^t} \mathbb{N}(\mu_h = 0, \sigma_h, \theta(h^t, p))$

$P(z_i^t|s^t) = \text{TBD}$

ME: How to deal with varying cluster sizes?

1) Use only mean instead of product.

2) Weight each sample by the percentage of the total cluster size by raising the resultant probability to the $\frac{\textbf{Total \# number particles}}{\textbf{\# Particles in Cluster}}$