Introduction

The filtering problem can be formulated as

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{v}_{k-1} \tag{1a}$$
$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{w}_k \tag{1b}$$

where $\mathbf{v}$ and $\mathbf{w}$ are the process noise and the observation noise. The state transition density is fully specified by $\mathbf{f}$ and the process noise distribution and the observation likelihood is fully specified by $\mathbf{h}$ and the observation noise distribution.

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = p_v(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1})) \tag{2a}$$
$$p(\mathbf{z}_k|\mathbf{x}_k) = p_w(\mathbf{z}_k - \mathbf{h}(\mathbf{x}_k)) \tag{2b}$$

The problem is to find an update formula from $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ to $p(\mathbf{x}_k|\mathbf{z}_{1:k})$, where $\mathbf{z}_{1:k}$ denotes all observation $\{\mathbf{z}_1, \ldots, \mathbf{z}_k\}$ up to time k. The Bayesian approach gives the following update:

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k) \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \mathrm{d}\mathbf{x}_{k-1}}{\int p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) \mathrm{d}\mathbf{x}_k} \tag{3}$$

Breaking the problem up, performing the multi-dimensional integration:

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \mathrm{d}\mathbf{x}_{k-1} \tag{4}$$

can be seen as stage one, and multiplying with $p(\mathbf{z}_k|\mathbf{x}_k)$ as stage two. In general the integral in stage one can not be calculated analytically, hence, we need some way of estimating the distribution $p(\mathbf{x}_k|\mathbf{z}_{1:k-1})$.

Kernel representation

With a Parzen density estimator a distribution can be approximated arbitrarily close by a number of identical kernels centered on points chosen from the distribution. In the particle filter the kernels are delta functions, but information can be gained by using a broader kernel. The distribution at time $k-1$ can be approximated by:

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \approx \sum_i^N w_{k-1}^i K(\mathbf{A}_{k-1}^i(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^i)) \tag{5}$$

where $\mathbf{A}^i$ is a transformation matrix used to keep track of distortions of the kernel. Each kernel can be propagated through the mapping $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ by using a local linearization, yielding a continuous output distribution $p(\mathbf{x}_k|\mathbf{z}_{1:k})$. This is again a sum of kernels but the kernels are no longer identical (in the sense that they are from the same family of functions, yet they have different parameters).

Parzen Particle Filters

Tue Lehn-Schiøler
Informatics and
Mathematical Modelling
Technical University of Denmark
tls@imm.dtu.dk

Deniz Erdogmus &
Jose C. Principe
CNEL
University of Florida
{deniz,principe} @cnel.ufl.edu

Using a Parzen density estimator the idea of particle filtering is extended. Kernels are propagated through the non linear functions instead of delta functions. In this way a better density estimate can be obtained and the number of particles can be reduced.

The Method

Using the kernel representation equation (4) can be written as:

$$\sum_i^N w_{k-1}^i \int p_v(\mathbf{x}_k - f(\mathbf{x}_{k-1})) K(\mathbf{A}_{k-1}^i(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^i)) \mathrm{d}\mathbf{x}_{k-1} \tag{6}$$

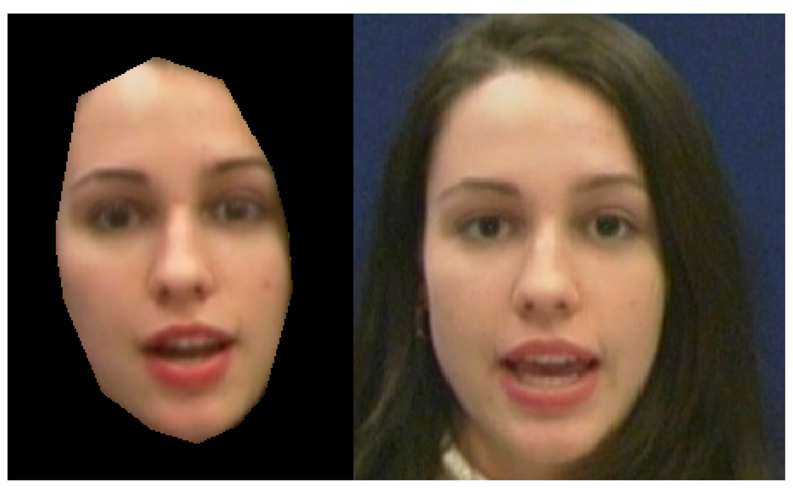
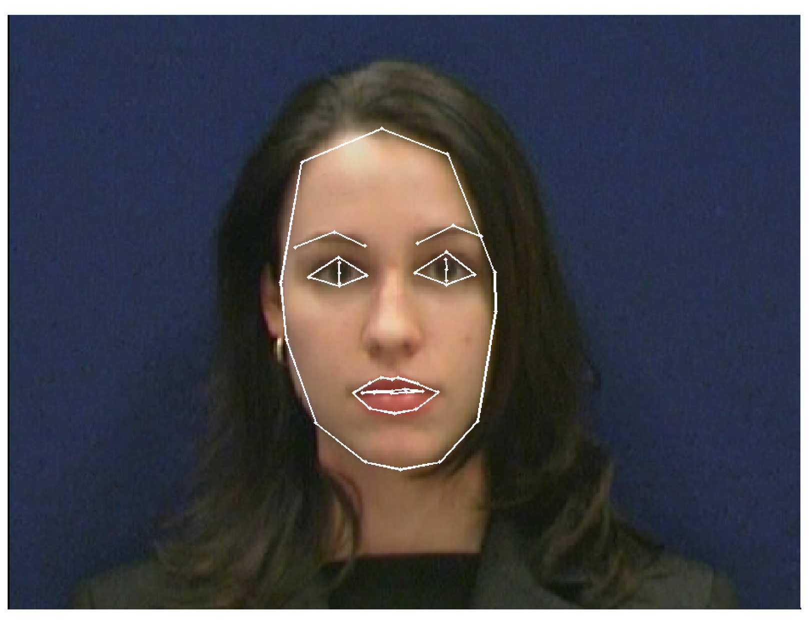
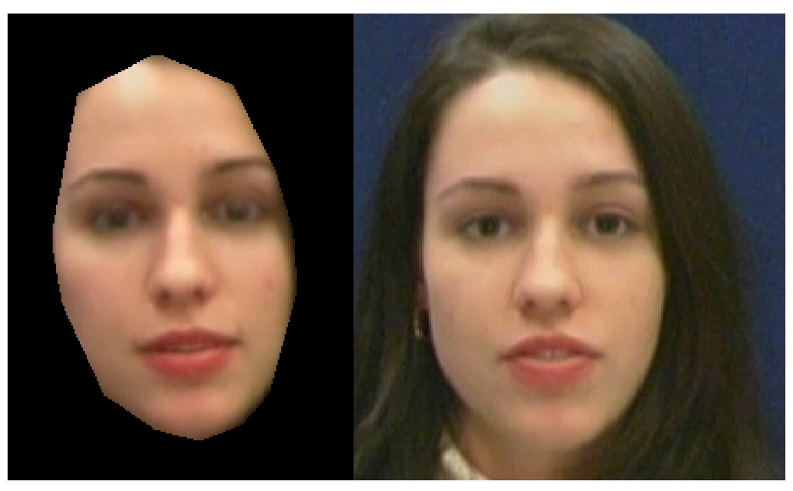
Each part of the sum can be handled individually and $\mathbf{f}$ can be locally linearized. By linearizing $\mathbf{f}$ around $\mathbf{x}_{k-1}^i$ the jacobian $\mathbf{J}|_{\mathbf{x}_{k-1}^i} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}|_{\mathbf{x}_{k-1}^i}$ is introduced and the following change of variables can be employed: $\hat{\mathbf{x}}_{k-1} = \mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}^i) - \mathbf{J}|_{\mathbf{x}_{k-1}^i}(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^i)$. Inserting this in the integral from equation (6) yields:

$$\left|\mathbf{J}|_{\mathbf{x}_{k-1}^i}\right|^{-1} \int [p_v(\hat{\mathbf{x}}_{k-1}) K\left(\mathbf{A}_{k-1}^i \mathbf{J}|_{\mathbf{x}_{k-1}^i}^{-1}(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}^i) - \hat{\mathbf{x}}_{k-1})\right)] \mathrm{d}\hat{\mathbf{x}}_{k-1} \tag{7}$$

This integral is an expectation over the process noise $E_{p_v}\left[K\left(\mathbf{A}_{k-1}^i \mathbf{J}|_{\mathbf{x}_{k-1}^i}^{-1}(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}^i) - \hat{\mathbf{x}}_{k-1})\right)\right]$ and can be approximated by a sample mean. In the extreme case a single sample drawn from p_v can be used, and the result is a translation of the kernel by the noise sample:

$$E_{p_v}\left[K\left(\mathbf{A}_{k-1}^i \mathbf{J}|_{\mathbf{x}_{k-1}^i}^{-1}(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}^i) - \hat{\mathbf{x}}_{k-1})\right)\right] \approx K\left(\mathbf{A}_{k-1}^i \mathbf{J}|_{\mathbf{x}_{k-1}^i}^{-1}(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}^i) - \mathbf{v}_{k-1})\right), \mathbf{v}_{k-1} \sim p_v$$

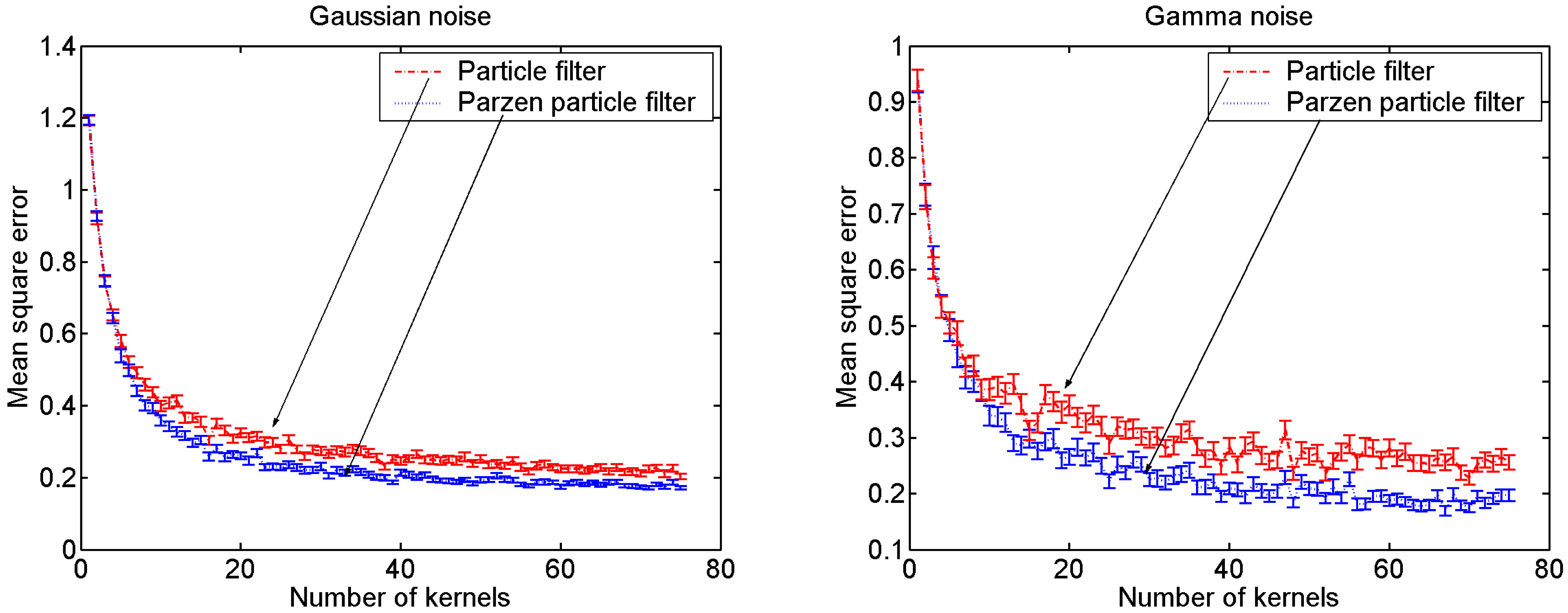
Writing $p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \sum_i^N w_k^i K(\mathbf{A}_k^i(\mathbf{x}_k - \mathbf{x}_k^i))$ It is possible to identify the centers of the kernels $\mathbf{x}_k^i = \mathbf{f}(\mathbf{x}_{k-1}^i) + \mathbf{v}_{k-1}$ and the transformation matrix $\mathbf{A}_k^i = \mathbf{A}_{k-1}^i \mathbf{J}|_{\mathbf{x}_{k-1}^i}^{-1}$. By considering equations (4,6,7) the weight update can be found to be $w_k^i = w_{k-1}^i p(\mathbf{z}_k|\mathbf{x}_k^i) \left|\mathbf{J}|_{\mathbf{x}_{k-1}}\right|^{-1}$. This derivation holds for any kernel.

Although not directly related, these images are examples of how state space models similar to the one from equation (1) can be used to predict mouth movements from a speech signal.
See e.g. www.imm.dtu.dk/∼tls/code/facedemo.php . The videos are taken from the VidTimit database.

Results

Performance of the Parzen particle filter is compared to the performance of the standard particle filter. The method is tested on a one dimensional problem. The mean square error is plotted as a function of the number of kernels, it can be seen that with few kernels the methods produces similar results, but as the number of kernels increases the kernel method becomes better. For this one dimensional example the number of particles can be reduced drastically by improving the density estimate.



ICASSP 2004, Montreal, May, 2004. Paper # 2669

Code to reproduce the results can be found at http://www.imm.dtu.dk/∼tls/code/code.php