Fast nonparametric clustering with Gaussian blurring mean-shift

Miguel Á. Carreira-Perpiñán
Dept. of Computer Science & Electrical Engineering, OGI/OHSU

http://www.csee.ogi.edu/~miguel
Gaussian mean-shift (GMS)

Given dataset \( X = \{x_n\}_{n=1}^N \subset \mathbb{R}^D \), define a Gaussian kernel density estimate with bandwidth \( \sigma \):

\[
p(x) = \frac{1}{N} \sum_{n=1}^{N} K\left( \frac{\|x - x_n\|}{\sigma} \right)
\]

\( K(t) = e^{-t/2} \).

GMS is an iterative algorithm to find a mode of \( p \): iterate

\( x^{(\tau+1)} = f(x^{(\tau)}) \)

with

\[
f(x) = \sum_{n=1}^{N} p(n|x)x_n
\]

\[
p(n|x) = \frac{e^{-\frac{1}{2} \frac{\|x - x_n\|^2}{\sigma^2}}}{\sum_{n'=1}^{N} e^{-\frac{1}{2} \frac{\|x - x_{n'}\|^2}{\sigma^2}}}.
\]

We can derive this from the stationary point equation \( \nabla p(x) = 0 \) as a fixed-point iteration.
Gaussian mean-shift (GMS) (cont.)

Paths followed by GMS for various starting points:
Gaussian mean-shift (GMS) (cont.)

- It has no step size.
- It is an EM algorithm with global, linear convergence (Carreira-Perpiñán & Williams ’03):
  - **E step**: update $p(n|x)$.
  - **M step**: update $x$.

Convergence can be sublinear or superlinear in some cases.

- It can be extended to arbitrary covariances and weights, and to other kernels (non-Gaussian).
- It is based on ideas by Fukunaga & Hostetler ’75 (also Cheng ’95, Carreira-Perpiñán ’00, Comaniciu & Meer ’02, etc.).
GMS as a clustering algorithm:

- $x_n, x_m$ in same cluster if they converge to same mode
- nonparametric clustering, able to deal with complex cluster shapes; $\sigma$ determines the number of clusters
- popular in computer vision (segmentation, tracking) (Comaniciu & Meer)

Segmentation example (one data point $x = (i, j, I)$ per pixel, where $(i, j) =$ location and $I = $ intensity or colour):
Pseudocode: GMS clustering

for $n \in \{1, \ldots, N\}$
    \[ x \leftarrow x_n \]
repeat
    \[ p(n|x) \leftarrow \frac{\exp \left( -\frac{1}{2} \| (x - x_n) / \sigma \|^2 \right)}{\sum_{n'=1}^{N} \exp \left( -\frac{1}{2} \| (x - x_{n'}) / \sigma \|^2 \right)} \]
    \[ x \leftarrow \sum_{n=1}^{N} p(n|x) x_n \]
until $x$'s update $< \text{tol}$
\[ z_n \leftarrow x \]
end
connected-components($\{z_n\}_{n=1}^{N}, \text{min\_diff}$)

For each data point
Starting point
Iteration loop
Post. prob. (E step)
Update $x$ (M step)
Mode
Clusters
Gaussian blurring mean-shift (GBMS)

- This is the algorithm that Fukunaga & Hostetler really proposed.
- It has gone largely unnoticed.
- Same iterative scheme, but now the data points move at each iteration. Result: sequence of progressively shrunk datasets $X^{(0)}$, $X^{(1)}$, ... converging to single, all-points-coincident cluster.
Pseudocode: GBMS clustering

repeat
  for $m \in \{1, \ldots, N\}$
    $\forall n$: $p(n | x_m) \leftarrow \frac{\exp \left( -\frac{1}{2} \| (x_m - x_n) / \sigma \|^2 \right)}{\sum_{n'=1}^{N} \exp \left( -\frac{1}{2} \| (x_m - x_{n'}) / \sigma \|^2 \right)}$
  $y_m \leftarrow \sum_{n=1}^{N} p(n | x_m) x_n$
end

$\forall m$: $x_m \leftarrow y_m$
until stop

connected-components($\{x_n\}_{n=1}^{N}, \text{min}\_\text{diff}$)

Iteration loop
For each data point

One GMS step
Update whole dataset
Stopping criterion
Clusters
Stopping criterion for GBMS

- GBMS converges to an all-points-coincident cluster (Cheng ’95). Proof: diameter of data set decreases at least geometrically.

- However, GBMS shows two phases:
  - **Phase 1**: points merge into clusters of coincident points (a few iterations); we want to stop here.
  - **Phase 2**: clusters keep approaching and merging (a few to hundreds of iterations); slowly erases clustering structure.

- We need a **stopping criterion** (as opposed to a **convergence criterion**) to stop just after phase 1.

- Simply checking $\|X^{(\tau)} - X^{(\tau-1)}\| < \text{tol}$ does not work because the points are always moving.

- Instead, consider the **histogram of updates** $\{e_n^{(\tau)}\}_{n=1}^N$, $e_n^{(\tau)} = \|x_n^{(\tau)} - x_n^{(\tau-1)}\|$. Though the histograms change as points move, in phase 2 the entropy $H$ does not change (the histogram bins change their position but not their values).
### Stopping criterion for GBMS (cont.)

<table>
<thead>
<tr>
<th>$\tau = 0$</th>
<th>$\tau = 1$</th>
<th>$\tau = 2$</th>
<th>$\tau = 3$</th>
<th>$\tau = 4$</th>
<th>$\tau = 5$</th>
<th>$\tau = 6$</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\tau = 7$</th>
<th>$\tau = 8$</th>
<th>$\tau = 9$</th>
<th>$\tau = 10$</th>
<th>$\tau = 11$</th>
<th>$\tau = 12$</th>
<th>$\tau = 13$</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image8.png" alt="Image" /></td>
<td><img src="image9.png" alt="Image" /></td>
<td><img src="image10.png" alt="Image" /></td>
<td><img src="image11.png" alt="Image" /></td>
<td><img src="image12.png" alt="Image" /></td>
<td><img src="image13.png" alt="Image" /></td>
<td><img src="image14.png" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\tau = 14$</th>
<th>$\tau = 15$</th>
<th>$\tau = 16$</th>
<th>$\tau = 17$</th>
<th>$\tau = 18$</th>
<th>$\tau = 19$</th>
<th>$\tau = 20$</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image15.png" alt="Image" /></td>
<td><img src="image16.png" alt="Image" /></td>
<td><img src="image17.png" alt="Image" /></td>
<td><img src="image18.png" alt="Image" /></td>
<td><img src="image19.png" alt="Image" /></td>
<td><img src="image20.png" alt="Image" /></td>
<td><img src="image21.png" alt="Image" /></td>
</tr>
</tbody>
</table>
Stopping criterion for GBMS (cont.)

Stopping criterion:

\[
\left( |H(e^{(\tau+1)}) - H(e^{(\tau)})| < 10^{-8} \right) \text{ OR } \left( \frac{1}{N} \sum_{n=1}^{N} e^{(\tau+1)}_n < \text{tol} \right)
\]
Convergence rate of GBMS

GBMS shrinks a Gaussian cluster towards its mean with cubic convergence rate. Proof:

\[ p(x) = \frac{1}{N} \sum_{n=1}^{N} K\left(\left\| \frac{x - x_n}{\sigma} \right\|^2 \right) \Rightarrow p(x) = \int_{\mathbb{R}^D} q(y) K(x - y) \, dy \]

New dataset:

\[ \tilde{x} = \int_{\mathbb{R}^D} p(y|x) y \, dy = \mathbb{E} \{y|x\} \quad \forall x \in \mathbb{R}^D \text{ with } p(y|x) = \frac{K(x - y)q(y)}{p(x)} \]

Considering 1D w.l.o.g.:

\[ q(x) = \mathcal{N}(x; 0, s^2) \Rightarrow p(x) = \mathcal{N}(x; 0, s^2 + \sigma^2), \quad q(\tilde{x}) = \mathcal{N}(\tilde{x}; 0, (rs)^2) \]

with \( r = \frac{1}{1+(\sigma/s)^2} \in (0, 1) \).
Convergence rate of GBMS (cont.)

Thus: \[ s^{(\tau+1)} = \frac{1}{1 + \left( \sigma/s^{(\tau)} \right)^2} s^{(\tau)} \]

for \( s^{(0)} > 0 \)

which converges to 0 with cubic order, i.e., \( \lim_{\tau \to \infty} \frac{|s^{(\tau+1)} - s^{(\infty)}|}{|s^{(\tau)} - s^{(\infty)}|^3} < \infty \).

Effectively, \( \sigma \) increases at every step.
In summary, a Gaussian distribution remains Gaussian with the same mean but each principal axis decreases cubically. This explains the practical behaviour shown by GBMS:

1. Clusters collapse extremely fast (clustering).
2. After a few iterations only the local principal component survives $\Rightarrow$ temporary linearly-shaped clusters (denoising).
Convergence rate of GBMS (cont.)

Number of GBMS iterations $\tau$ necessary to achieve $s^{(\tau)} < \text{tol}$ as a function of the bandwidth $\sigma$ for $s^{(0)} = 1$:

![Graph showing the convergence rate](image)

Note that GMS converges with linear order ($p = 1$), thus requiring many iterations to converge to a mode.
Connection with spectral clustering

GBMS pseudocode (inner loop)

\[
\text{for } m \in \{1, \ldots, N\} \\
\forall n: p(n|x_m) \leftarrow \frac{\exp\left(-\frac{1}{2} \left\|(x_m - x_n) \right\|^2 / \sigma^2\right)}{\sum_{n'=1}^{N} \exp\left(-\frac{1}{2} \left\|(x_m - x_{n'}) \right\|^2 / \sigma^2\right)} \\
y_m \leftarrow \sum_{n=1}^{N} p(n|x_m)x_n \\
\text{end} \\
\forall m: x_m \leftarrow y_m
\]

Same pseudocode in matrix form

\[
W = \left(\exp\left(-\frac{1}{2} \left\|(x_m - x_n) \right\|^2 / \sigma^2\right)\right)_{nm} \\
D = \text{diag}\left(\sum_{n=1}^{N} w_{nm}\right) \\
X = X WD^{-1}
\]

For each data point

One GMS step

Update whole dataset

Gaussian affinity matrix

Degree (normalising) matrix

Update whole dataset
GBMS can be written as repeated products \( X \leftarrow X P \) with the random-walk matrix \( P = WD^{-1} \) equivalent to the graph Laplacian (\( W \): Gaussian affinities, \( D \): degree matrix, \( P \): posterior probabilities \( p(n|x_m) \)).

In phase 1 \( P \) is quickly changing as points cluster.

In phase 2 \( P \) is almost constant (and perfectly blocky) so GBMS implicitly extracts the leading eigenvectors (power method) like spectral clustering.

Thus GBMS is much faster than computing eigenvectors (about 5 matrix-vector products are enough).

Actually, since \( P \) is a positive matrix it has a single leading eigenvector (Perron-Frobenius th.) with constant components, so eventually all points collapse.
Connection with spectral clustering (cont.)

Leading 7 eigenvectors $u_1, \ldots, u_7 \in \mathbb{R}^N$ and leading 20 eigenvalues $\mu_1, \ldots, \mu_{20} \in (0, 1]$ of $P = (p(n|x_m))_{nm}$:

<table>
<thead>
<tr>
<th>$\tau$</th>
<th>$u_2$</th>
<th>$u_3$</th>
<th>$u_4$</th>
<th>$u_5$</th>
<th>$u_6$</th>
<th>$u_7$</th>
<th>$\mu_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td><img src="image1" alt="Image" /></td>
<td><img src="image2" alt="Image" /></td>
<td><img src="image3" alt="Image" /></td>
<td><img src="image4" alt="Image" /></td>
<td><img src="image5" alt="Image" /></td>
<td><img src="image6" alt="Image" /></td>
<td><img src="graph1" alt="Graph" /></td>
</tr>
<tr>
<td>2</td>
<td><img src="image7" alt="Image" /></td>
<td><img src="image8" alt="Image" /></td>
<td><img src="image9" alt="Image" /></td>
<td><img src="image10" alt="Image" /></td>
<td><img src="image11" alt="Image" /></td>
<td><img src="image12" alt="Image" /></td>
<td><img src="graph2" alt="Graph" /></td>
</tr>
<tr>
<td>4</td>
<td><img src="image13" alt="Image" /></td>
<td><img src="image14" alt="Image" /></td>
<td><img src="image15" alt="Image" /></td>
<td><img src="image16" alt="Image" /></td>
<td><img src="image17" alt="Image" /></td>
<td><img src="image18" alt="Image" /></td>
<td><img src="graph3" alt="Graph" /></td>
</tr>
<tr>
<td>6</td>
<td><img src="image19" alt="Image" /></td>
<td><img src="image20" alt="Image" /></td>
<td><img src="image21" alt="Image" /></td>
<td><img src="image22" alt="Image" /></td>
<td><img src="image23" alt="Image" /></td>
<td><img src="image24" alt="Image" /></td>
<td><img src="graph4" alt="Graph" /></td>
</tr>
<tr>
<td>11</td>
<td><img src="image25" alt="Image" /></td>
<td><img src="image26" alt="Image" /></td>
<td><img src="image27" alt="Image" /></td>
<td><img src="image28" alt="Image" /></td>
<td><img src="image29" alt="Image" /></td>
<td><img src="image30" alt="Image" /></td>
<td><img src="graph5" alt="Graph" /></td>
</tr>
</tbody>
</table>
Accelerated GBMS algorithm

- At each iteration, replace clusters already formed with a single point of mass = no. points in cluster.
- The algorithm is now an alternation between GMS blurring steps and connected-component reduction steps.
- Equivalent to the original GBMS but faster.
- The effective number of points $N^{(\tau)}$ (thus the computation) decreases very quickly.

**Computational cost:**
- $k_1$: average number of iterations per point for GMS
- $k_2$: number of iterations for GBMS and accelerated GBMS

<table>
<thead>
<tr>
<th></th>
<th>GMS</th>
<th>GBMS</th>
<th>Accelerated GBMS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$2N^2 D k_1$</td>
<td>$\frac{3}{2} N^2 D k_2$</td>
<td>$\frac{3}{2} D \sum_{\tau=1}^{k_2} (N^{(\tau-1)})^2$</td>
</tr>
</tbody>
</table>
Experiments with image segmentation

Original image

<table>
<thead>
<tr>
<th>Image</th>
<th>GMS</th>
<th>GBMS</th>
<th>Accelerated GBMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>cameraman 124 × 124</td>
<td>71.5 ($\sigma = 24.2$)</td>
<td>18 ($\sigma = 20.3$)</td>
<td>4.6 ($\sigma = 20.3$)</td>
</tr>
<tr>
<td>hand 137 × 110</td>
<td>36.4 ($\sigma = 33$)</td>
<td>14 ($\sigma = 24$)</td>
<td>4.8 ($\sigma = 24$)</td>
</tr>
</tbody>
</table>
Experiments with image segmentation (cont.)

Effective number of points $N^{(\tau)}$ in accelerated GBMS:

Accelerated GBMS is

- $2 \times - 4 \times$ faster than GBMS
- $5 \times - 60 \times$ faster than GMS
Experiments with image segmentation (cont.)

GMS

GBMS and accelerated GBMS

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>$C$</th>
<th>Speedup</th>
<th>Iterations</th>
<th>Clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.4</td>
<td>6</td>
<td>8</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>10.1</td>
<td>4</td>
<td>8</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>18.3</td>
<td>2</td>
<td>8</td>
<td>5</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>$C$</th>
<th>Speedup</th>
<th>Iterations</th>
<th>Clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1</td>
<td>6</td>
<td>8</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>8.8</td>
<td>4</td>
<td>8</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>15.8</td>
<td>2</td>
<td>8</td>
<td>5</td>
<td>2</td>
</tr>
</tbody>
</table>
Experiments with image segmentation (cont.)

- Segmentations of similar quality to those of GMS but faster.
- Computational cost $O(kN^2)$. Possible further accelerations:
  - Fast Gauss transform
  - Sparse affinities $W$:
    - Truncated Gaussian kernel
    - Proximity graph ($k$-nearest-neighbours, etc.)
- Bandwidth $\sigma$ set by the user; good values of $\sigma$:
  - GMS: $\sigma \approx \frac{1}{5} \times$ (image-side)
  - GBMS: somewhat smaller
Can also try values of $\sigma$ in a scaled-down version of the image (fast) and then scale up $\sigma$. 
Conclusion

- We have turned a neglected algorithm originally due to Fukunaga & Hostetler into a practical algorithm by introducing a reliable stopping criterion and an acceleration.
- Fast convergence (cubic order for Gaussian clusters).
- Connection with spectral clustering (GBMS interleaves refining the affinities with power iterations).
- Excellent segmentation results, faster than GMS and spectral clustering; only parameter is $\sigma$ (which controls the granularity).
- Very simple to implement.
- We hope to see it more widely applied.