Fast Evaluation of Kernel Distances

Yaseen Syed & Andreas Mang
Department of Mathematics, University of Houston, Houston, TX, USA

UNIVERSITY of HOUSTON

Introduction

Teaser: Our goal is the analysis and implementation of fast computational methods for the evaluation of kernel distances in the context of shape matching and shape classification.

Kernel-based methods play a key role in many applications of computational sciences and engineering, and—in particular—in machine learning. A major drawback of kernel methods is their computational complexity and scalability. If implemented naively, they require linear storage, with a cubic computational complexity for direct methods, and a quadratic complexity for iterative methods. In the present work, we consider methods for the evaluation of sums of Gaussian kernels with a linear complexity; the considered class of algorithms is referred to **Fast Gauss Transform** (**FGT**) [1,2,3,4,5].

Shape Matching and Shape Classification

In the present work, we consider kernel-based methods in the context of shape matching and shape classification [6,7]. Examples of different shapes in $\mathbb{R}^d$ for $d = 3$ are shown in **Figure 1**.
In a nutshell, the shape matching problem is as follows: Given two shapes s_0 and s_1 included in $\mathbb{R}^d$, we seek a time-dependent $\mathbb{R}^d$-diffeomorphism $\boldsymbol{f}$ that is found by solving

$$\underset{\boldsymbol{v},\,\boldsymbol{f}}{\text{minimize}} \quad \text{dist}(\boldsymbol{f}(s_0), s_1) + \text{reg}(\boldsymbol{v}). \tag{1}$$

The flow of the $\mathbb{R}^d$-diffeomorphism $\boldsymbol{f}$ is the solution of the flow equation $\mathrm{d}_t\boldsymbol{f} = \boldsymbol{v}(\boldsymbol{f})$ for $t \in (0,1]$ with initial condition $\boldsymbol{f} = \mathrm{id}_{\mathbb{R}^d}$ for $t = 0$. The functional dist models the discrepancy between the deformed shape $\boldsymbol{f}(s_0)$ and the target shape s_1. The control variable is a time-dependent velocity field $\boldsymbol{v}$. The functional reg is a regularization model that prescribes sufficient regularity on $\boldsymbol{v}$ to ensure that $\boldsymbol{f}$ is a diffeomorphism [6,7].

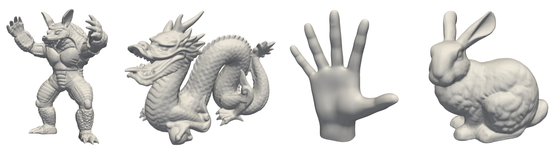

Figure 1: *Illustration of various shape representations in $\mathbb{R}^3$.*

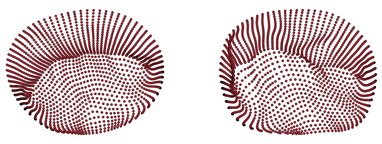

Figure 2: *Shapes of cardiac mitral valves of two patients represented as point clouds $\boldsymbol{x} \in \mathbb{R}^{3n}$ and $\boldsymbol{y} \in \mathbb{R}^{3m}$.*

To make the problem computationally tractable, we consider a self-reproducing kernel Hilbert space formulation. In this setting, we model the shapes s_0 and s_1 discretized by n and m points $\boldsymbol{x} := (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \in \mathbb{R}^{3n}$ and $\boldsymbol{y} := (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m) \in \mathbb{R}^{3m}$, respectively (see **Figure 2** for an illustration). Based on this representation, the shape distance in (1) is modeled as a kernel distance

$$\begin{aligned}\text{dist}(\boldsymbol{y}, \boldsymbol{x}) = &\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{n^2}\kappa(\boldsymbol{y}_i, \boldsymbol{y}_j) - \sum_{i=1}^{n}\sum_{j=1}^{m}\frac{2}{nm}\kappa(\boldsymbol{y}_i, \boldsymbol{x}_j)\\ &+ \sum_{i=1}^{m}\sum_{j=1}^{m}\frac{1}{m^2}\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)\end{aligned} \tag{2}$$

with $\kappa(\boldsymbol{y}_i, \boldsymbol{x}_j) \propto \exp(-\|\boldsymbol{y}_i - \boldsymbol{x}_j\|_2^2/\sigma^2)$. In what follows, we consider fast kernel-based method to enable a quick evaluation of (2).

Fast Gauss Transform

The discrete Gauss transform is given by

$$\phi(\boldsymbol{y}_k) = \sum_{i=1}^{n} c_i \kappa(\boldsymbol{y}_k, \boldsymbol{x}_i) \tag{3}$$

with Gaussian kernel $\kappa(\boldsymbol{y}_k, \boldsymbol{x}_i) = \exp(-\|\boldsymbol{y}_k - \boldsymbol{x}_i\|_2^2/\sigma^2)$, weights $\{c_i \in \mathbb{R}\}_{i=1}^n$, source points $\{\boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^n$ (centers of the Gaussian kernels), target points $\{\boldsymbol{y}_i \in \mathbb{R}^d\}_{i=1}^m$, and bandwidth $\sigma > 0$. A direct evaluation of (3) for m target points with n source points has a complexity of $\mathcal{O}(mn)$. The FGT [1,2] is an ε-exact algorithm with nominal linear complexity $\mathcal{O}(m+n)$. The key idea of the FGT is to expand the kernel $\kappa : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ into Hermite functions. For a univariate Gaussian kernel $\kappa : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ we obtain

$$\exp(-(y-x)^2/\sigma^2) \approx \sum_{k=0}^{p-1}\frac{1}{k!}\Big(\frac{x - x_*}{\sigma}\Big)^k h_k\Big(\frac{y - x_*}{\sigma}\Big) \tag{4a}$$

$$\exp(-(y-x)^2/\sigma^2) \approx \sum_{k=0}^{p-1}\frac{1}{k!}\Big(\frac{y - x_*}{\sigma}\Big)^k h_k\Big(\frac{x - x_*}{\sigma}\Big) \tag{4b}$$

with expansion center $x_* \in \mathbb{R}$ and univariate Hermite functions $h_k : \mathbb{R} \to \mathbb{R}$,

$$h_k(x) := (-1)^k \frac{\mathrm{d}^k}{\mathrm{d}x^k}\exp(-x^2).$$

Notice that the expansions in (4) are identical, except for the fact that the arguments of the Hermite functions and the monomial functions are flipped. To evaluate the FGT for source and target points in a d-dimensional space, we can treat the multivariate Gaussian function as a Kronecker product of d univariate Gaussian functions, applying the series expansions (4) to each dimension. Let $\boldsymbol{\alpha} := (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}^d$ denote a multi-index with length $|\boldsymbol{\alpha}| = \alpha_1 + \alpha_2 + \cdots + \alpha_d$ and factorial $\boldsymbol{\alpha}! = \alpha_1!\alpha_2!\cdots\alpha_d!$. With this, we can represent (3) using the expansion

$$\phi(\boldsymbol{y}_k) = \sum_{\boldsymbol{\alpha} \succeq 0} \rho_{\boldsymbol{\alpha}} h_{\boldsymbol{\alpha}}\Big(\frac{\boldsymbol{y}_k - \boldsymbol{x}_*}{\sigma}\Big)$$

about the center point $\boldsymbol{x}_* \in \mathbb{R}^d$, with multidimensional Hermite functions $h_{\boldsymbol{\alpha}} = \prod_{i=1}^d h_{\alpha_i}(x_i)$ and coefficients $\rho_{\boldsymbol{\alpha}}$ defined by

$$\rho_{\boldsymbol{\alpha}} = \frac{1}{\boldsymbol{\alpha}!}\sum_{i=1}^{n} c_i\Big(\frac{\boldsymbol{x}_i - \boldsymbol{x}_*}{\sigma}\Big)^{\boldsymbol{\alpha}},$$

where $\boldsymbol{x}^{\boldsymbol{\alpha}} = \prod_{i=1}^d x_i^{\alpha_i}$ is a d-variate moninomial of degree $|\boldsymbol{\alpha}|$ for any $\boldsymbol{x} \in \mathbb{R}^d$.
A key limitation of the traditional FGT algorithm [1,2] is that the constant in the complexity estimate $\mathcal{O}(m+n)$ grows exponentially with increasing dimensionality d; in particular, if we truncate the Hermite expansion after p terms, the complexity is $\mathcal{O}((m+n)p^d)$. The truncation number p controls the accuracy of the approximation (i.e., $\varepsilon > 0$). The fact that the complexity of the algorithm grows exponentially with d makes the traditional FGT algorithm [1,2] computationally prohibitive in high-dimensional statistics and pattern recognition applications. An implementation that overcomes some of the limitations of the traditional FGT is the **improved FGT** (**IFGT**) [3,4,5]. We consider this implementation here.

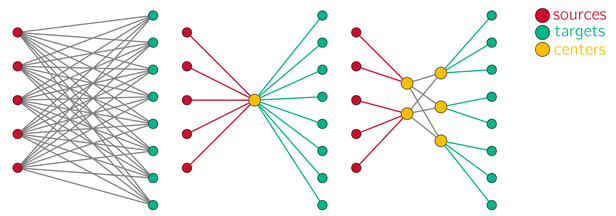

Figure 3: *Left: Direct evaluation of Gauss transform for m target points with n sources with a complexity of $\mathcal{O}(mn)$. Middle & right: The FGT reduces this complexity by constructing truncated p-term series about a small number of source cluster-centers resulting in $\mathcal{O}(np^d)$ operations. These series are subsequently shifted to target cluster centers, and evaluated at m targets, resulting in $\mathcal{O}(mp^d)$ operations.*

Results

We study the performance of the IFGT algorithm [3,4,5] for random point clouds $\boldsymbol{x} \in \mathbb{R}^{3n}$ and $\boldsymbol{y} \in \mathbb{R}^{3m}$ in a unit cube $[0,1]^3$. We consider an increasing number of target and source points (with $n = m$). We report error estimates (with respect to the direct evaluation), runtime times, and the associated speedup in **Table 1**.

Table 1: *Comparison of computational performance between the IFGT [3,4,5] and the direct evaluation of the Gauss transform for a varying number of target and source points (with $m = n$), where $d = 3$, $\sigma = 0.02$, and the target error is set to $\varepsilon = 1\mathrm{e}{-2}$. We report (from left to right) the number of target and source points, the error between direct and approximate evaluation of the Gauss transform, the runtime (in seconds) for the direct evaluation with nominal complexity $\mathcal{O}(mn)$ and the IFGT [3,4,5] with nominal complexity $\mathcal{O}(m+n)$, and the speedup attained for the IFGT.*

$n\,(=m)$	error	direct	FGT	speedup
1 000	2.28e−5	2.01e−2	3.97e−3	5.05
5 000	8.50e−6	4.73e−1	1.65e−2	28.65
10 000	7.57e−6	1.92	3.19e−2	60.12
50 000	3.37e−6	4.74e1	3.33e−1	142.24
100 000	2.80e−6	1.85e2	1.22	151.66

References

1. L. Greengard & J. Strain. The fast Gauss transform. SIAM Journal on Scientific and Statistical Computing, 12(1):79–94, 1991.
2. J. Strain. The fast Gauss transform with variable scales. SIAM Journal on Scientific and Statistical Computing, 12(5):1131–1139, 1991.
3. C. Yang, R. Duraiswami & L. Davis. Efficient Kernel Machines Using the Improved Fast Gauss Transform. In Advances in Neural Information Processing Systems, Volume 17, pp. 1561–1568, 2005.
4. C. Yang, R. Duraiswami, N. Gumerov & L. Davis. Improved fast Gauss transform and efficient kernel density estimation. In Proc. ICCV 2003, pages 464–471, 2003.
5. C. Yang, R. Duraiswami & N. A. Gumerov. Improved fast gauss transform. Technical Report CS-TR-4495, UMIACS, University of Maryland, College Park, 2003.
6. P. Zhang, A. Mang, J. He, R. Azencoot, K.-C. El Tallawi & W. A. Zoghbi. Diffeomorphic Shape Matching by Operator Splitting in 3D Cardiology Imaging, Journal of Optimization Theory and Applications 188(1):143–168, 2021.
7. R. Azencott et al. Diffeomorphic matching and dynamic deformable surfaces in 3D medical imaging. Computational Methods in Applied Mathematics, 10(3):235–274, 2010.