Sub-sampling for Efficient Spectral Mesh Processing

Rong Liu, Varun Jain, and Hao Zhang

GrUVi Lab, School of Computing Sciences, SFU, BC, Canada
{lrong, vjain, haoz}@cs.sfu.ca

Abstract. In this paper, we apply Nyström method, a sub-sampling and reconstruction technique, to speed up spectral mesh processing. We first relate this method to Kernel Principal Component Analysis (KPCA). This enables us to derive a novel measure in the form of a matrix trace, based soly on sampled data, to quantify the quality of Nyström approximation. The measure is efficient to compute, well-grounded in the context of KPCA, and leads directly to a greedy sampling scheme via trace maximization. On the other hand, analyses show that it also motivates the use of the max-min farthest point sampling, which is a more efficient alternative. We demonstrate the effectiveness of Nyström method with farthest point sampling, compared with random sampling, using two applications: mesh segmentation and mesh correspondence.

1 Introduction

Spectral methods for data modeling and processing have been well studied in machine learning and pattern recognition, e.g., for clustering [1, 2] and correspondence analysis [3, 4]. The idea is to derive, from relational data given as a matrix and typically of high dimensionality, a low-dimensional and information-preserving spatial embedding based on the eigenvectors of the matrix, to facilitate the processing or analysis task at hand. Recently, spectral techniques have been applied successfully to several mesh processing problems, including spectral decomposition for mesh compression [5], spectral clustering for mesh segmentation [6], 3D shape correspondence in the spectral domain [7], spectral sequencing for mesh streaming [8], segmentation [9], and as an aid to surface reconstruction [10], as well as surface flattening via multidimensional scaling [11].

One of the main drawbacks of spectral methods is that they can be computationally expensive for large data sets since they rely on eigenvector computation and at times also require a non-sparse matrix, whose construction involves determining pairwise affinities between a large number of points. Nyström approximation [12], a sub-sampling and reconstruction technique originated from integral calculus, has been proposed as a remedy, e.g., for image segmentation [13], but there lacks a formal analysis of its quality and the influence of the sampling procedure. So far, random sampling [13, 14] has been used predominantly.

In this paper, we cast Nystöm approximation in the context of kernel PCA (KPCA), where the samples are treated as training data. The ability of the

training set to capture the probabilistic distribution of the whole data set in the feature space induces a way to measure the quality of the Nyström method. The resulting measure can be derived as a matrix trace, which depends only on sampled data. This measure is more desirable than the Schur complement [13], the only known quality measure for Nyström so far. Empirically, we show that both measures produce consistent evaluation results. Furthermore, our novel quality measure is more efficient to compute and leads directly to a greedy sampling scheme via trace maximization. On the other hand, analyses of the measure show that it motivates the use of a more efficient heuristic sampling scheme, which turns out to be the *max-min farthest point sampling*.

The rest of the paper is organized as follows. After discussing previous work, we describe Nyström approximation and spectral embedding in Section 3. KPCA is briefly reviewed in Section 4. Relating Nyström to KPCA, we propose our quality measure for Nyström in Section 5. We then discuss the relevance of our quality measure to sampling and motivate the use of the farthest point scheme. In Section 7, we demonstrate experimentally the effectiveness of the Nyström method and farthest point sampling, compared to random sampling, using two applications. Finally, we conclude and comment on possible future work.

2 Previous work

The graph Laplacian operator has been well studied in geometry processing, e.g., see the recent survey [15]. In particular, the Fiedler vector, eigenvector of the graph Laplacian corresponding to the second smallest eigenvalue, has been used in graph partitioning [16] and mesh sequencing [8]. For the planar mesh graph, the Laplacian is sparse, for which fast multilevel methods, e.g., ACE [17], can compute the leading eigenvectors efficiently. However, when many eigenvectors are needed, e.g., for spectral mesh compression [5], the cost would be too high for large data sets. In this case, the mesh is often partitioned into smaller pieces and the operation proceeds in a piecewise manner [5].

Problems such as surface flattening [11], mesh segmentation [6, 9], shape correspondence [7], and most instances of clustering and dimensionality reduction considered in the machine learning and pattern recognition literature, e.g., [1, 2], rely on more global relational information. In these cases, an affinity matrix is defined by applying a Gaussian-like filter to a distance matrix H, where H_{ij} is a suitably defined distance, e.g., Euclidean [1], geodesic [7, 11], graph distance, or a combination of them [6, 9], between points i and j in a data set. Computing the full affinity matrix takes quadratic time and since it is generally non-sparse, it is computationally expensive to obtain its eigenvectors. Nyström approximation has been proposed recently to speed up spectral methods in this case [13, 18].

The Nyström method only requires a small number of sampled rows of the affinity matrix. It solves a small-scale eigenvalue problem and then computes approximated eigenvectors via extrapolation. One of the main questions is how to design appropriate sampling schemes to obtain more accurate approximations of the ground-truth eigenvectors. To the best of our knowledge, this problem has not been studied before. So far, random sampling predominates [13, 14] and other

simple schemes, e.g., max-min farthest point sampling [14], have been mentioned in passing, but with no analysis given.

3 Nyström approximation and spectral embeddings

Applications [16, 7, 2] utilizing spectral embeddings start by building a matrix which encodes certain relationship, called affinities, between each pair of elements in a data set. Depending on the application (see Section 7), this matrix may be transformed and then its eigenvectors and possibly eigenvalues are used to obtain a spatial embedding of the original data points in the spectral domain. Computing spectral embeddings is time-consuming due to the quadratic complexity, in terms of the data size, of affinity computation and up to cubic-time complexity for eigenvalue decomposition. Nyström method [13, 18] is therefore proposed to overcome this problem via sub-sampling and reconstruction.

Consider a set of n points $\mathcal{Z} = \mathcal{X} \bigcup \mathcal{Y}$, where $\mathcal{X}$ and $\mathcal{Y}$, $\mathcal{X} \bigcap \mathcal{Y} = \varnothing$, are two subsets of size l and m. Write the symmetric affinity matrix $W \in \mathbb{R}^{n \times n}$ in block form $W = [A \;\; B; B^T \;\; C]$, where $A \in \mathbb{R}^{l \times l}$ and $C \in \mathbb{R}^{m \times m}$ are affinity matrices for points in $\mathcal{X}$ and $\mathcal{Y}$, respectively; $B \in \mathbb{R}^{l \times m}$ contains the *cross-affinities* between points in $\mathcal{X}$ and $\mathcal{Y}$. Without loss of generality, we designate the points in $\mathcal{X}$ as *sample points*. Let $A = U \Lambda U^T$ be the eigenvalue decomposition of A, then the eigenvectors of W can be approximated, using the Nyström method [13], as

$$\bar{U} = \begin{bmatrix} U \\ B^T U \Lambda^{-1} \end{bmatrix}. \tag{1}$$

This allows us to approximate the eigenvectors of W by only knowing the sampled sub-block $[A \;\; B]$. The overall complexity is thus reduced from $O(n^3)$, without sub-sampling, down to $O(ml^2) + O(l^3)$, where $l \ll n$, in practice.

The rows of $\bar{U}$ define the spectral embeddings of the original data points from $\mathcal{Z}$. From (1), we see that the i^{th} row of U, which is completely determined by A, gives the embedding $\bar{x}_i$ of point x_i in $\mathcal{X}$ and the j^{th} row of $B^T U \Lambda^{-1}$ is the embedding $\bar{y}_j$ of point y_j in $\mathcal{Y}$. If we let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_l$ be the eigenvalues of A, and $\bar{y}_j^d$ denote the d^{th} component of $\bar{y}_j$, then equation (1) can be rewritten as

$$\bar{y}_j^d = \frac{1}{\lambda_d} \sum_{i=1}^{l} \bar{x}_i^d B(i,j) = \frac{1}{\lambda_d} \sum_{i=1}^{l} \bar{x}_i^d W(i, j+l), \qquad 1 \leq d \leq l. \tag{2}$$

Namely, the embedding $\bar{y}_j$ is extrapolated using the coordinates of the $\bar{x}_i$'s, weighted by the corresponding cross-affinities in B.

With $\bar{U}$, we obtain an approximation $\bar{W}$ of the original affinity matrix W,

$$\bar{W} = \bar{U} \Lambda \bar{U}^T = \begin{bmatrix} A & B \\ B^T & B^T A^{-1} B \end{bmatrix}.$$

Clearly, $\bar{W}$ replaces block C of W with $B^T A^{-1} B$. Hence it is suggested to quantify the approximation quality using the norm of the *Schur complement*, $C - B^T A^{-1} B$. The smaller the norm, the better the approximation. Schur complement has been used as the *de facto* quality measure for Nyström method [13].

4 Review of kernel PCA

Suppose that the points in set $\mathcal{Z}$ lie in the Euclidean space $\mathbb{R}^g$. Kernel PCA (KPCA) [19], an extension to the standard PCA, first applies to $\mathcal{Z}$ a generally non-linear mapping $\phi : \mathbb{R}^g \to \mathcal{F}$, where $\mathcal{F}$ is referred to as the *feature space*. Then the standard PCA is carried out in $\mathcal{F}$ on the point set $\phi(\mathcal{Z}) = \{\phi(z_i)|z_i \in \mathcal{Z}\}$. Since $\mathcal{F}$ may have a very high, possibly infinite, dimensionality, the non-linear properties of the data $\mathcal{Z}$ can be "unfolded" into linear ones. Thus algorithms that work on linear structures, e.g., PCA, can be effectively applied in $\mathcal{F}$.

The mapping ϕ is never explicitly given, but implicitly specified by the inner products between the data and encoded in a *kernel matrix* $K \in \mathbb{R}^{n\times n}$, where $K_{ij} = k(z_i, z_j) = \phi(z_i)\cdot\phi(z_j)$. Algorithms running in the feature space based only on inner products can be efficiently executed in the original space by replacing inner products with the kernel function k. Gaussian radial basis function [19]

$$k(z_i, z_j) = e^{-\frac{d_{ij}^2}{2\sigma^2}}, \;\; d_{ij} = ||z_i - z_j||^2. \tag{3}$$

is one of the most commonly used kernels. Note that in our applications, we set σ to the average of all distances computed. Assume that $\mathcal{Z}$ obeys a certain probability distribution and $\mathcal{X} \subseteq \mathcal{Z}$ is chosen as a *training set* and centered. Although ϕ is not known explicitly, it is still possible to compute the projections $\tilde{x}_i$, the *features*, of $\phi(x_i)$ into the space where the basis are the principal components of the point set $\phi(\mathcal{X}) = \{\phi(x_i)|x_i \in \mathcal{X}\}$, as follows. Let $L \in \mathbb{R}^{l\times l}$ be the upper-left block of K and $L = E\Lambda E^T$ the eigenvalue decomposition of L, where eigenvectors, the columns of E, are in descending eigenvalue order. Let e_r denote the r^{th} row of E, then the d^{th} coordinate of $\tilde{x}_i$ is given by $\tilde{x}_i^d = \frac{1}{\sqrt{\lambda_d}} \sum_{r=1}^{l} e_r^d k(x_r, x_i)$. This can be seen as a "black box", which returns the feature $\tilde{x}_i$ for any given point x_i. If the training set $\mathcal{X}$ characterizes the distribution well, it is then reasonable to apply it to $y_j \in \mathcal{Y}$ similarly,

$$\tilde{y}_j^d = \frac{1}{\sqrt{\lambda_d}} \sum_{r=1}^{l} e_r^d k(x_r, y_j) = \frac{1}{\sqrt{\lambda_d}} \sum_{r=1}^{l} e_r^d K(r, j+l). \tag{4}$$

It can be seen that the embedding of $\tilde{y}_j$ in the feature space can be constructed using the eigenvector entries of the sub-kernel L, weighted by the kernel entries defined between the x_i's and the y_j's. Here it is worth noting that a resemblance between KPCA and Nyström approximation (2) is emerging.

5 Quality measure for Nyström approximation

While the affinity matrix W defines spectral embeddings and the kernel K is used in KPCA, both matrices can be seen as an implicit definition of relations between data points. Equating W with K and comparing equation (2) with (4), we see that $\bar{y}_j^d$ and $\tilde{y}_j^d$ have essentially identical expressions up to a scaling factor $\lambda_d^{-1/2}$; this fact has been previously noted in [18] as well. Therefore, Nyström approximation can be considered as a process of running KPCA on new patterns through a training set. In this section, we investigate the approximation quality of Nyström method in the context of KPCA.

5.1 Quality measure as a matrix trace

Considering the Nyström method in the context of KPCA, we treat the sample set $\mathcal{X}$ as the training set. Thus points in $\mathcal{X}$ are first mapped into the feature space $\mathcal{F}$ and then the features corresponding to points in $\mathcal{Y}$ are approximated. In this setting, a good training set $\mathcal{X}$ for accurate Nyström approximation should be a set that, as stated in Section 4, reflects the same probability distribution function as $\mathcal{Y}$. This implies that $\phi(\mathcal{Z})$ and $\phi(\mathcal{X})$ should roughly lie in the same space. To this end, we stipulate that an accurate Nyström approximation necessitates that the sum of squared distances from all the $\phi(z_i)$'s to the space spanned by the $\phi(x_i)$'s be small. Now we derive our quality measure.

Denote by Σ the covariance matrix of the point set $\phi(\mathcal{X})$. Note that although the $\phi(x_i)$'s can be high-dimensional, the rank of Σ can be no larger than $l = |\phi(\mathcal{X})|$ and Σ has at most l eigenvectors $(\xi_1, \xi_2, \dots \xi_l)$, corresponding to non-zero eigenvalues. Let $P = [\xi_1|...|\xi_l]$, then the squared distance from $\phi(z_i)$ to the space spanned by the $\phi(x_i)$'s (equivalently, the column space of P) is

$$\rho_i = ||\phi(z_i) - PP^T\phi(z_i)||^2 = ||\phi(z_i)||^2 - ||PP^T\phi(z_i)||^2,$$

where PP^T is the *orthogonal projection operator* which projects any vector into the column space of P. Hence we wish to minimize the objective function

$$\begin{aligned}\sum_{i=1}^{n}\rho_i &= \sum_{z_i\in\mathcal{X}}\rho_i + \sum_{z_j\in\mathcal{Y}}\rho_j \\ &= \left(\sum_{i=1}^{l}||\phi(x_i)||^2 - \sum_{i=1}^{l}||PP^T\phi(x_i)||^2\right) + \left(\sum_{j=1}^{m}||\phi(y_j)||^2 - \sum_{j=1}^{m}||PP^T\phi(y_j)||^2\right).\end{aligned}$$

Note that $\sum_{i=1}^{l}||\phi(x_i)||^2 + \sum_{j=1}^{m}||\phi(y_j)||^2 = \sum_{i=1}^{n}||\phi(z_i)||^2 = \sum_{i=1}^{n}K_{ii}$. Also, $\sum_{i=1}^{l}||PP^T\phi(x_i)||^2 = \sum_{i=1}^{l}||\phi(x_i)||^2 = \sum_{i=1}^{l}K_{ii}$. For our purpose, K is derived using the Gaussian kernel (3), thus the diagonals of K are constant 1. As a result, the first three terms of the objective function are constant, given that the size of $\mathcal{X}$ is fixed. Our goal is then reduced to *maximizing* the quantity

$$\Gamma = \sum_{j=1}^{m}||PP^T\phi(y_j)||^2 = \sum_{j=1}^{m}(P^T\phi(y_j))^T(P^T\phi(y_j)). \tag{5}$$

Denote by $U_1, U_2, \dots, U_l$ the eigenvectors of A, as first defined in Section 3. Note that since we now make no difference between K and W, A is also the upper-left block of K. It is known [19] the principal component ξ_i of $\phi(\mathcal{X})$ can be written as a linear combination of the $\phi(x_i)$'s, i.e. $\xi_i = \sum_{d=1}^{l}\lambda_i^{-\frac{1}{2}}U_i^d\phi(x_d)$. We then have $\xi_i^T\phi(y_j) = \sum_{d=1}^{l}\lambda_i^{-\frac{1}{2}}U_i^d[\phi(x_d)\cdot\phi(y_j)] = \sum_{d=1}^{l}\lambda_i^{-\frac{1}{2}}U_i^d k(x_d, y_j)$. Therefore

$$P^T\phi(y_j) = [\xi_1|\dots|\xi_l]^T\phi(y_j) = [\xi_1^T\phi(y_j)|\dots|\xi_l^T\phi(y_j)]^T = (U\Lambda^{-\frac{1}{2}})^T B_j,$$

where B_j is the j-th column of B. Thus (5) is simplified to

$$\Gamma = \sum_{j=1}^{m}B_j^T U\Lambda^{-1}U^T B_j = \mathrm{tr}(B^T A^{-1}B), \tag{6}$$

where $\mathrm{tr}(\cdot)$ denotes the matrix trace. When Γ attains a larger value, in the feature space, points in $\mathcal{Z}$ lie closer to the space spanned by $\mathcal{X}$. Consequently, Nyström method achieves a better approximation. Thus Γ provides a quality measure for Nyström method. The time complexity for computing Γ is $O(ml^2) + O(l^3)$. In practice, $l \ll n$ and can be regarded as a constant. Throughout our experiments (Section 7), we set $l = 10$. Next, we empirically verify the accuracy of Γ against Schur complement and then use it to derive a heuristic sampling scheme.

5.2 Comparison with Schur complement

Typically, the norm of the Schur complement $C - B^T A^{-1} B$, defined in Section 3, is used to measure the approximation quality of Nyström. But the time complexity involved would be $O(n^2)$, since C is required. For efficiency, C should never be computed fully; this excludes the possibility of using the Schur complement on the fly as a measure to supervise the sampling process. On the other hand, Γ does not suffer from this problem and is much more efficient to compute.

Quality-wise, let us use the Schur complement as the ground truth to evaluate the accuracy of Γ, empirically. In each run of our experiment, a set of points is generated using a Gaussian distribution. From these points, two sample sets S_1 and S_2 of equal size are randomly chosen. We check whether the two measures would rank the two sample sets consistently. Denote by $\eta(S_i)$ and $\gamma(S_i)$ the approximation error values resulting from using the norm of the Schur complement and Γ, respectively, on sample set S_i. We plot, as a red marker, the position of the 2D point $(\eta(S_1) - \eta(S_2), \gamma(S_2) - \gamma(S_1))$. Obviously, the marker would appear in the first or third quadrant if and only if the two approaches had produced the same ranking of the sample sets. In Fig. 1, results from 100 test runs are shown. Evidently, almost all points lie in the first and third quadrants. This verifies, experimentally, the robustness of Γ as a quality measure for Nyström approximation.

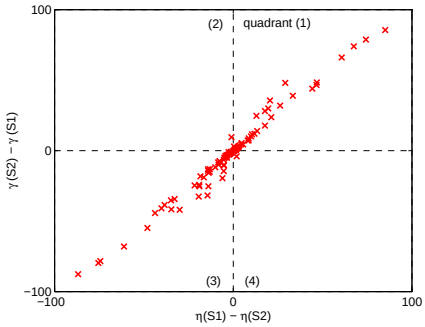


Fig. 1. Only 4% of the red markers lie in the second and fourth quadrants. This shows the consistence between Γ and the norm of the Schur complement, in evaluating sampling schemes for Nyström approximation.

6 Sampling scheme

Depending on the application, different sampling strategies for Nyström method may be considered. But so far random sampling [13, 14] has been the norm. Random sampling may work well when the sample size is sufficiently large. But large sample size will increase the workload. In the applications we discuss in Section 7, we wish to take few samples while still achieving good performance.

6.1 Greedy sampling based on Γ

As verified in Section 5.2, Γ provides a good quality measure for Nyström method, which can be used *on the fly* to guide a *greedy sampling procedure*. Specifically, each time a new sample is added, its affinities to the remaining points are maintained. This way we always know the current A and B, the within-sample and cross-sample affinities. The next sample is the one which if selected, the updated B and A would maximize Γ.

However, it is worth pointing out that we cannot afford to evaluate Γ on all the un-sampled points, which makes finding one sample a procedure with a complexity of at least $O(m^2)$. Instead, we resort to the *best candidate sampling* scheme. Namely, we do not evaluate Γ on all the remaining points to locate the best one, but only to find a sample among the best 1% with a probability of 95%. Assuming that the remaining points are independent identically distributed random variables, we only need to search for the best sample (maximizing Γ) from a random subset of size $\lceil \log(0.01)/\log(0.95) \rceil = 90$ regardless of the problem size. Our experiments show that this greedy scheme works quite well and efficient since a very low sampling rate can be used. Moreover, it can be made more efficient without computing Γ explicitly, as we show below.

6.2 The max-min farthest point sampling scheme

To further speed up the greedy sampling scheme without sacrificing much of its quality, we propose to use a heuristic which would not require explicit computation of Γ. This is made possible by examining the mathematical properties of Γ. Based on the cyclic property of matrix trace operation, we know that

$$\Gamma = \mathrm{tr}(B^T A^{-1} B) = \mathrm{tr}(A^{-1} B B^T) = \mathrm{tr}(A^{-1} \sum_{j=1}^{m} B_j B_j^T).$$

When m is large, entries of $M = \sum_{j=1}^{m} B_j B_j^T$ are close to each other. If we rewrite $M \approx \tau \mathbf{1}\mathbf{1}^\mathbf{T}$, with $\tau \in (0, m)$ a fixed value, then $\Gamma \approx \tau \mathrm{tr}(A^{-1}\mathbf{1}\mathbf{1}^\mathbf{T}) = \tau \mathbf{1}^\mathbf{T}(\mathbf{A^{-1}1})$. Observe that there are two conditions for Γ to attain a relatively large value. The first is to have a large $\mathbf{1}^T(A^{-1}\mathbf{1})$. Note that $A^{-1}\mathbf{1}$ gives the coefficients of the expansion of $\mathbf{1}$ in the space whose basis are the columns of A. Moreover, the diagonals of A are 1 and its other entries lie in $(0, 1)$. It is easy to show, in 2D, that the sum of these coefficients, $\mathbf{1}^T(A^{-1}\mathbf{1})$, is no larger than l. The maximal l is obtained when A's columns are the canonical basis of the Euclidean space. This is generalizable to arbitrary dimensions. In order for A's columns to be close to the canonical basis, the off-diagonal entries should be close to zero. Thus samples should be taken *mutually far away* from each other.

The second condition is to have a larger τ. To this end, entries of B should be large, meaning that the distances from the samples to the remaining points should be small on average. When a sufficient number of sample points are distributed mutually far away, the average distances from the remaining points to the sample points tend to converge, making this condition much less influential. Our experiments also verify that the first condition plays a dominant role.

Stimulated by the first condition, we propose to use the more efficient max-min farthest point sampling scheme for Nyström method, which works as follows:

1. Randomly pick a point q and find p which is farthest away. Switch p and q and repeat. After several iterations, p is chosen as the first sample s_1. This procedure will most likely place s_1 close to an extremity of the point set.
2. At step i, a new sample s_i is chosen as the one which maximizes the minimum distance (hence "max-min") to the previous samples $s_1, s_2, \ldots, s_{i-1}$.

The significance of having the quality measure Γ is that it induces a greedy sampling scheme, which dramatically speeds up the spectral embedding as a very low sampling rate can be used. Moreover, it provides an underlying motivation for using farthest point sampling at an even lower computational cost.

7 Applications

Now we apply Nyström method with max-min farthest point sampling to two applications and evaluate the results both numerically and visually.

7.1 Mesh correspondence

A recent variant to the classical mesh parameterization problem is to compute a *cross parameterization* [20] between two meshes directly, where mesh correspondence, computed over sets of selected features on the two meshes, is often the first step. Currently, most methods [20] rely on manual feature selection and correspondence. Spectral techniques have been proposed in the past to compute correspondence between feature points on two 2D images [4, 21] and they can be applied to find the much needed initial mapping between mesh features as well [7]. In this section, we apply Nyström method to a simple 3D extension of one such spectral correspondence algorithm by Shapiro and Brady [21]. Specifically, pairwise similarities between data points are given by the L_2 distances between their spectral embeddings and best matching is used to recover correspondence.

Instead of using Euclidean distances [21] to define the affinities, we use geodesic point-to-point distances on the meshes to better handle articulated shapes. We also make two modifications to the original algorithm as follows. First we use only the leading k eigenvectors of the affinity matrix to compute the spectral embeddings and secondly, we scale the eigenvectors with the square root of the corresponding eigenvalues. Both modifications have been shown to improve the correspondence in the case of 3D meshes [7].

To visualize the correspondence, we use color coding of vertices. If X and Y are the two meshes to be matched, we first assign colors to every vertex in Y; we carefully assign colors so that different parts of the mesh are colored differently. Then, we set the color of every vertex of X to be the color of the matching vertex in Y. Thus a good correspondence induces similar coloring in the two shapes. Also shown is a comparison between random and farthest point sampling. Clearly, the matching obtained using Nyström approximation with farthest point

(a) (b)

Fig. 2. Results from spectral mesh correspondence. (a) Top row: ground truth, without sampling. The next two rows show results using Nyström, with farthest-point and random sampling, respectively. Inconsistent coloring at badly matched points for the latter are highlighted by circles. (b) Results for larger meshes (4000 faces) using Nyström and farthest point sampling. Shapes on the left are matched with those on the right. As we can see, with only 10 samples, we already obtain excellent correspondence results.

sampling is comparable to the ground truth, which is the matching computed via eigen-decomposition of the full affinity matrix. Fig. 2(b) shows more correspondence results obtained using Nyström approximation on larger meshes (4000 triangles), with the same k and sample size.

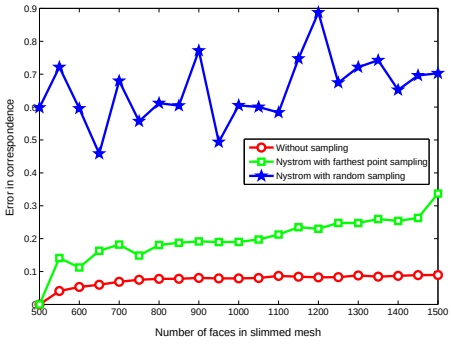


Fig. 3. Plot of correspondence error against mesh size. Nyström with farthest point sampling has comparable performance to the un-sampled case.

Note that results shown in Fig. 2(a) and 2(b) are subjective. Evaluating a dense correspondence objectively is non-trivial since the ground-truth correspondence is not known and is impractical to establish manually for large data sets. To present a more objective evaluation, we use the following trick. We first construct a series of decimated meshes using QSLIM [22]. Then we find the correspondence between the original mesh and a decimated version. Since QSLIM does

not alter the position of un-decimated vertices, the ground-truth correspondence can be trivially computed from a decimated mesh. The correspondence error at a vertex is defined as the geodesic distance between the found matching point for the vertex and its ground-truth matching point. In Fig. 3, the total error is plotted for all vertices against the size of the mesh. This plot is averaged over several meshes. Again, it can be seen that Nyström method, when combined with farthest sampling, has comparable performance to its much more costly counterpart, where the full affinity matrices are used and the eigenvectors are accurately computed.

7.2 Mesh segmentation

In part-type mesh segmentation [23], the goal is to decompose a mesh shape into its constituent components according to human intuition. Since mesh segmentation can be considered as a problem of clustering mesh faces, spectral clustering becomes applicable. In the work of [6], spectral embeddings of faces are first derived from the intrinsic geometric property of the shape, followed by a K-means clustering in the spectral embedding space. The rationale behind this approach is that face clusters in the embedding space correspond to parts of the shape. In [6], sub-sampling is not conducted and all pairwise distances have to be computed and converted into affinities. Subsequently, the eigenvectors of the affinity matrix are computed to find the face embeddings. Although it is possible to lower the workload by computing only the leading eigenvectors, it is still prohibitive for large meshes, since computing pairwise distances alone would take $O(n^2 \log n)$ time for an n-face mesh.

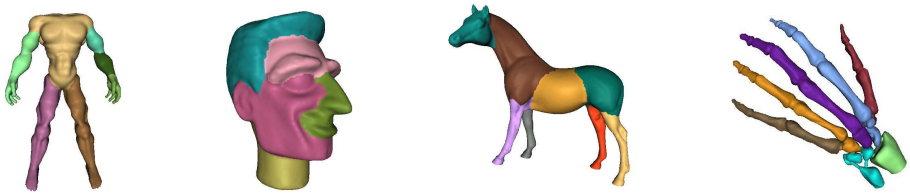
Alternatively, we can apply max-min farthest point sampling and Nyström method to approximate the spectral embeddings of faces. Supposing that the sample size is l, we only need to compute the distances from the l sample faces to the remaining faces since only the sub-block $[A \;\; B]$ of W is needed. The whole process for computing the embeddings then takes $O(ln \log n)$ time. Since $l \ll n$, the computational overhead is dramatically reduced.

Fig. 4 presents several segmentation results using Nyström method with farthest point sampling, where parts are indicated by different colors. As we can see, the segmentation results are quite intuitive even at a very low sample rate of 10. Table 1 reports the timing. Compared with the results in [6], which only handles meshes of size up to 4000 faces, in about 30 seconds, the improvement is quite evident.

In Fig. 5, we compare the performance of Nyström method under random and farthest point sampling. It is easy to see that Nyström method works much better with farthest point sampling. Also shown in this figure is that a better sampling (indicated by a larger Γ value) leads to a more meaningful segmentation. For the two pictures, (b) and (d), produced using Nyström and farthest point sampling, no visually differences from those obtained in [6] can be observed.

Table 1. Statistics of segmentation experiments on a 2.2 GHz Pentium machine with 1.0 GB RAM. Note that since iterative 2-means, instead of a single K-means, is used on the horse and hand bone models, their running time is relatively higher.

Model	**Heart**	**Igea**	**Headless**	**Smile**	**Horse**	**Hand bone**
Face #	1619	2000	32,574	34,712	39,698	65,001
Part #	4	3	7	5	8	7
Time (s)	0.03	0.07	2.23	3.32	6.86	9.67



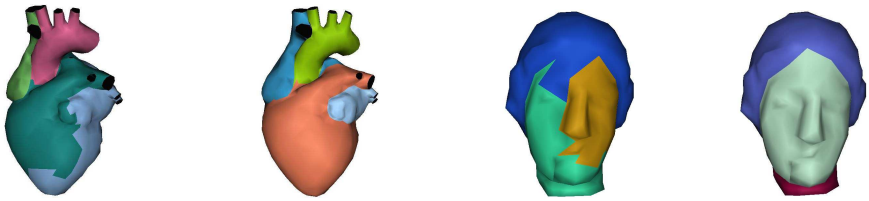
(a) Headless (b) Smile (c) Horse (d) Hand bone

Fig. 4. Segmentation results. We test the effectiveness of Nyström method for both K-means (a, b) and iterative 2-means (c, d). Farthest point sampling is used.

8 Conclusion and future work

In this work, we study the approximation quality of Nyström approximation, an important approach for speeding up kernel based algorithms. To overcome the difficulty of investigating the method directly, we cast it in the context of KPCA. With the help of the geometric intuition offered by the KPCA framework, a simple yet accurate quality measure for the Nyström method is derived. This quality measure can be used on the fly to guide a greedy sampling process for better approximation. To improve efficiency, we analyze its mathematical properties and motivate the use of the max-min farthest point sampling scheme. We apply Nyström method and farthest point sampling to two mesh processing algorithms, correspondence and segmentation, to demonstrate their effectiveness. At the same time, we also experiment with applying the same framework to spectral sequencing with positive results achieved. But due to limited space, we shall report those results elsewhere.

One possible future work is to consider in more detail the relationship between KPCA and Nyström method when various preprocessing procedures are applied to the affinity matrix. Another improvement is to study how different kernels, such as Gaussian, exponential kernel and polynomial kernels, would influence the behaviors of the Nyström method. It is also interesting to come up with application-based evaluation for the effectiveness of Γ. With mesh segmentation as an example, it is desirable to be able to measure the segmentation quality quantitatively so that the approximation performance of Nyström method can be evaluated based on the final result directly.



(a) Heart [0.75] (b) Heart [1.06] (c) Igea [0.636] (d) Igea [0.99]

Fig. 5. Comparison of segmentation results under different sampling schemes. (a, c) are results when random sampling is taken; (b, d) are obtained using farthest point sampling. The numbers in brackets are the Γ values divided by the number of faces.

References

1. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: analysis and an algorithm. In: NIPS. (2002) 857–864
2. Weiss, Y.: Segmentation using eigenvectors: a unifying view. In: ICCV. (1999) 975–982
3. Caelli, T., Kosinov, S.: An eigenspace projection clustering method for inexact graph matching. IEEE Trans. on PAMI **26**(4) (2004) 515–519
4. Carcassoni, M., Hancock, E.R.: Spectral correspondence for point pattern matching. Pattern Recognition **36** (2003) 193–204
5. Karni, Z., Gotsman, C.: Spectral compression of mesh geometry. In: SIGGRAPH 2000. (2000) 279–286
6. Liu, R., Zhang, H.: Segmentation of 3d meshes through spectral clustering. In: Proc. Pacific Graphics. (2004) 298–305
7. Jain, V., Zhang, H.: Robust 3d shape correspondence in the spectral domain. In: Shape Modeling International. (2006) to appear
8. Isenburg, M., Lindstrom, P.: Streaming meshes. In: IEEE Visualization. (2005) 231–238
9. Zhang, H., Liu, R.: Mesh segmentation via recursive and visually salient spectral cuts. In: Vision, Modeling, and Visualization. (2005)
10. Kolluri, R., Shewchuk, J.R., O'Brien, J.F.: Spectral surface reconstruction from noisy point clouds. In: Eurographics SGP. (2004) 11–21
11. Zigelman, G., Kimmel, R., Kiryati, N.: Texture mapping using surface flattening via multidimensional scaling. IEEE TVCG **8**(2) (2002) 198–207
12. Press, W., Tekolsky, S., Vetterling, W., Flannery, B.: Numerical Recipies in C. Cambridge Univ. Press (1992)
13. Fowlkes, C., Belongie, S., Chung, F., Malik, J.: Spectral grouping using the nyström method. IEEE Trans. PAMI **26** (2004) 214–225
14. de Silva, V., Tenenbaum, B.: Sparse multidimensional scaling using landmark points. Technical report, Stanford University (2004)
15. Sorkine, O.: Laplacian mesh processing. In: Eurographics State-of-the-Art Report. (2005)
16. Shi, J., Malik, J.: Normalized cuts and image segmentation. In: CVPR. (1997) 731–737
17. Harel, D., Koren, Y.: A fast multi-scale method for drawing large graphs. In: Graph Drawing. (2000) 183–196

18. Williams, C., Seeger, M.: Using the nyström method to speed up kernel machines. In: Advances in Neural Information Processing Systems. (2001) 682–688
19. Scholkopf, B., Smola, A., Muller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation **10** (1998) 1299–1319
20. Kraevoy, V., Sheffer, A.: Cross-parameterization and compatible remeshing of 3d models. In: ACM SIGGRAPH. (2004)
21. Shapiro, L.S., M., B.J.: Feature based correspondence: An eigenvector approach. Image and Vision Computing **10**(5) (1992) 283–288
22. Garland, M.: (Qslim simplification software — qslim 2.1)
23. Shamir, A.: A formalization of boundary mesh segmentation. In: Proc. 2nd Int'l Symposium on 3D Data Processing, Visualization, and Transmission. (2004) 82–89