We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



14

Computing and Updating Principal Components of Discrete and Continuous Point Sets

Darko Dimitrov
Freie Universität Berlin
Germany

1. Introduction

Efficient computation and updating of the principal components are crucial in many applications including data compression, exploratory data analysis, visualization, image processing, pattern and image recognition, time series prediction, detecting perfect and reflective symmetry, and dimension detection. The thorough overview over PCA's applications can be found for example in the textbooks by Duda et al. (2001) and Jolliffe (2002).

Dynamic versions of the PCA applications, i.e., when the point set (population) changes, are of big importance and interest. Efficient solutions of those problems depend heavily on an efficient dynamic computation of the principal components (eigenvectors of the covariance matrix). Dynamic updates of variances in different settings have been studied since the sixties by Chan et al. (1979), Knuth (1998), Pébay (2008), Welford (1962) and West (1979). Pébay (2008) also investigated the dynamic maintenance of covariance matrices.

The principal components of discrete point sets can be strongly influenced by point clusters (Dimitrov, Knauer, Kriegel & Rote (2009)). To avoid the influence of the distribution of the point set, often continuous sets, especially the convex hull of a point set is considered, which lead to so-called continuous PCA. Computing PCA bounding boxes (Gottschalk et al. (1996), Dimitrov, Holst, Knauer & Kriegel (2009)), or retrieval of 3D-objects (Vranić et al. (2001)), are typical applications where continuous PCA are of interest.

The organization and the main results presented in this chapter are as follows: In Section 2, we present a standard approach of computing principal components of discrete point set in $\mathbb{R}^d$. In Section 3, we present closed-form solutions for efficiently updating the principal components of a set of n points, when m points are added or deleted from the point set. For both operations performed on a discrete point set in $\mathbb{R}^d$, we can compute the new principal components in $O(m)$ time for fixed d. This is a significant improvement over the commonly used approach of recomputing the principal components from scratch, which takes $O(n+m)$ time. In Section 4 we consider the computation of the principal components of a dynamic continuous point set. We give closed form-solutions when the point set is a convex polytope $\mathbb{R}^3$. Solutions for the cases when the point set is the boundary of a convex polytope in $\mathbb{R}^2$ or $\mathbb{R}^3$, or a convex polygon in $\mathbb{R}^2$, are presented in the appendix. Conclusion and open problems are presented in Section 5.

For implementation and verification of some the theoretical results presented here, we refer interested readers to Dimitrov, Holst, Knauer & Kriegel (2009) and Dimitrov et al. (2011).

2. Basics. Computing principal components - discrete case in $\mathbb{R}^d$

The central idea and motivation of PCA is to reduce the dimensionality of a point set by identifying *the most significant directions (principal components)*. Let $P = \{\vec{p}_1, \vec{p}_2, \ldots, \vec{p}_n\}$ be a set of vectors (points) in $\mathbb{R}^d$, and $\vec{\mu} = (\mu_1, \mu_2, \ldots, \mu_d) \in \mathbb{R}^d$ be the center of gravity of P. For $1 \leq k \leq d$, we use $p_{i,k}$ to denote the k-th coordinate of the vector p_i. Given two vectors $\vec{u}$ and $\vec{v}$, we use $\langle \vec{u}, \vec{v} \rangle$ to denote their inner product. For any unit vector $\vec{v} \in \mathbb{R}^d$, the *variance of P in direction $\vec{v}$* is

$$\mathrm{var}(P, \vec{v}) = \frac{1}{n} \sum_{i=1}^{n} \langle p_i - \vec{\mu}\,,\, \vec{v} \rangle^2. \tag{1}$$

The most significant direction corresponds to the unit vector $\vec{v}_1$ such that $\mathrm{var}(P, \vec{v}_1)$ is maximum. In general, after identifying the j most significant directions $\vec{v}_1, \ldots, \vec{v}_j$, the $(j+1)$-th most significant direction corresponds to the unit vector $\vec{v}_{j+1}$ such that $\mathrm{var}(P, \vec{v}_{j+1})$ is maximum among all unit vectors perpendicular to $\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_j$.

It can be verified that for any unit vector $\vec{v} \in \mathbb{R}^d$,

$$\mathrm{var}(P, \vec{v}) = \langle \Sigma\, \vec{v}, \vec{v} \rangle, \tag{2}$$

where Σ is the *covariance matrix* of P. Σ is a symmetric $d \times d$ matrix where the (i, j)-th component, $\sigma_{ij}, 1 \leq i, j \leq d$, is defined as

$$\sigma_{ij} = \frac{1}{n} \sum_{k=1}^{n} (p_{i,k} - \mu_i)(p_{j,k} - \mu_j). \tag{3}$$

The procedure of finding the most significant directions, in the sense mentioned above, can be formulated as an eigenvalue problem. If $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$ are the eigenvalues of Σ, then the unit eigenvector $\vec{v}_j$ for λ_j is the j-th most significant direction. Since the matrix Σ is symmetric positive semidefinite, its eigenvectors are orthogonal, all λ_js are non-negative and $\lambda_j = \mathrm{var}(X, \vec{v}_j)$.

Computation of the eigenvalues, when d is not very large, can be done in $O(d^3)$ time, for example with the *Jacobi* or the *QR method* (Press et al. (1995)). Thus, the time complexity of computing principal components of n points in $\mathbb{R}^d$ is $O(n + d^3)$. The additive factor of $O(d^3)$ throughout the paper will be omitted, since we will assume that d is fixed. For very large d, the problem of computing eigenvalues is non-trivial. In practice, the above mentioned methods for computing eigenvalues converge rapidly. In theory, it is unclear how to bound the running time combinatorially and how to compute the eigenvalues in decreasing order. In Cheng & Y. Wang (2008) a modification of the *Power method* (Parlett (1998)) is presented, which can give a guaranteed approximation of the eigenvalues with high probability.

3. Updating the principal components efficiently - discrete case in $\mathbb{R}^d$

In this section, we consider the problem of updating the covariance matrix Σ of a discrete point set $P = \{\vec{p}_1, \vec{p}_2, \ldots, \vec{p}_n\}$ in $\mathbb{R}^d$, when m points are added or deleted from P. We give

closed-form solutions for computing the components of the new covariance matrix Σ'. Those closed-form solutions are based on the already computed components of Σ. The recent result of Pébay (2008) implies the same solution for additions that will be presented in the sequel. The main result of this section is given in the following theorem.

Theorem 3.1. *Let P be a set of n points in $\mathbb{R}^d$ with known covariance matrix Σ. Let P' be a point set in $\mathbb{R}^d$, obtained by adding or deleting m points from P. The principal components of P' can be computed in $O(m)$ time for fixed d.*

Proof. **Adding points**

Let $P_m = \{\vec{p}_{n+1}, \vec{p}_{n+2}, \ldots, \vec{p}_{n+m}\}$ be a point set with center of gravity $\vec{\mu}^m = (\mu_1^m, \mu_2^m, \ldots, \mu_d^m)$. We add P_m to P obtaining new point set P'. The j-th component, μ'_j, $1 \leq j \leq d$, of the center of gravity $\vec{\mu}' = (\mu'_1, \mu'_2, \ldots, \mu'_d)$ of P' is

$$\mu'_j = \frac{1}{n+m} \sum_{k=1}^{n+m} p_{k,j} = \frac{1}{n+m} \left(\sum_{k=1}^{n} p_{k,j} + \sum_{k=n+1}^{n+m} p_{k,j} \right) = \frac{n}{n+m}\mu_j + \frac{m}{n+m}\mu_j^m.$$

The (i,j)-th component, σ'_{ij}, $1 \leq i, j \leq d$, of the covariance matrix Σ' of P' is

$$\begin{aligned}\sigma'_{ij} &= \frac{1}{n+m} \sum_{k=1}^{n+m} (p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j) \\ &= \frac{1}{n+m} \sum_{=1}^{n} (p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j) + \frac{1}{n+m} \sum_{k=n+1}^{n+m} (p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j).\end{aligned}$$

Let

$$\sigma'_{ij} = \sigma'_{ij,1} + \sigma'_{ij,2},$$

where,

$$\sigma'_{ij,1} = \frac{1}{n+m} \sum_{k=1}^{n} (p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j), \tag{4}$$

and

$$\sigma'_{ij,2} = \frac{1}{n+m} \sum_{k=n+1}^{n+m} (p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j). \tag{5}$$

Plugging-in the values of μ'_i and μ'_j in (4), we obtain:

$$\begin{aligned}\sigma'_{ij,1} &= \frac{1}{n+m} \sum_{k=1}^{n} (p_{k,i} - \frac{n}{n+m}\mu_i - \frac{m}{n+m}\mu_i^m)(p_{k,j} - \frac{n}{n+m}\mu_j - \frac{m}{n+m}\mu_j^m) \\ &= \frac{1}{n+m} \sum_{k=1}^{m} (p_{k,i} - \mu_i + \frac{m}{n+m}\mu_i - \frac{m}{n+m}\mu_i^m)(p_{k,j} - \mu_j + \frac{m}{n+m}\mu_j - \frac{m}{n+m}\mu_j^m) \\ &= \frac{1}{n+m} \sum_{k=1}^{n} (p_{k,i} - \mu_i)(p_{k,j} - \mu_j) + \frac{1}{n+m} \sum_{k=1}^{n} (p_{k,i} - \mu_i)(\frac{m}{n+m}\mu_j - \frac{m}{n+m}\mu_j^m) + \\ &\quad \frac{1}{n+m} \sum_{k=1}^{n} (\frac{m}{n+m}\mu_i - \frac{m}{n+m}\mu_i^m)(p_{k,j} - \mu_j) + \\ &\quad \frac{1}{n+m} \sum_{k=1}^{n} (\frac{m}{n+m}\mu_i - \frac{m}{n+m}\mu_i^m)(\frac{m}{n+m}\mu_j - \frac{m}{n+m}\mu_j^m).\end{aligned}$$

Since $\sum_{k=1}^{n}(p_{k,i}-\mu_i)=0, 1\le i\le d$, we have

$$\sigma'_{ij,1}=\frac{n}{n+m}\sigma_{ij}+\frac{nm^2}{(n+m)^3}(\mu_i-\mu_i^m)(\mu_j-\mu_j^m). \tag{6}$$

Plugging-in the values of μ'_i and μ'_j in (5), we obtain:

$$\begin{aligned}
\sigma'_{ij,2} &= \frac{1}{n+m}\sum_{k=n+1}^{n+m}(p_{k,i}-\frac{n}{n+m}\mu_i-\frac{m}{n+m}\mu_i^m)(p_{k,j}-\frac{n}{n+m}\mu_j-\frac{m}{n+m}\mu_j^m)\\
&= \frac{1}{n+m}\sum_{k=n+1}^{n+m}(p_{k,i}-\mu_i^m+\frac{n}{n+m}\mu_i^m-\frac{n}{n+m}\mu_i)(p_{k,j}-\mu_j^m+\frac{n}{n+m}\mu_j^m-\frac{n}{n+m}\mu_j)\\
&= \frac{1}{n+m}\sum_{k=n+1}^{n+m}(p_{k,i}-\mu_i^m)(p_{k,j}-\mu_j^m)+\frac{1}{n+m}\sum_{k=n+1}^{n+m}(p_{k,i}-\mu_i^m)\frac{n}{n+m}(\mu_j^m-\mu_j)+\\
&\quad \frac{1}{n+m}\sum_{k=n+1}^{n+m}\frac{n}{n+m}(\mu_i^m-\mu_i)(p_{k,j}-\mu_j^m)+\\
&\quad \frac{1}{n+m}\sum_{k=n+1}^{n+m}\frac{n}{n+m}(\mu_i^m-\mu_i)\frac{n}{n+m}(\mu_j^m-\mu_j).
\end{aligned}$$

Since $\sum_{k=n+1}^{n+m}(p_{k,i}-\mu_i^m)=0, 1\le i\le d$, we have

$$\sigma'_{ij,2}=\frac{m}{n+m}\sigma_{ij}^m+\frac{n^2m}{(n+m)^3}(\mu_i-\mu_i^m)(\mu_j-\mu_j^m), \tag{7}$$

where

$$\sigma_{ij}^m=\frac{1}{m}\sum_{k=n+1}^{n+m}(p_{k,i}-\mu_i^m)(p_{k,j}-\mu_j^m)),\quad 1\le i,j\le d,$$

is the i,j-th element of the covariance matrix Σ_m of the point set P_m. Finally, we have

$$\sigma'_{ij}=\sigma'_{ij,1}+\sigma'_{ij,2}=\frac{1}{n+m}(n\sigma_{ij}+m\sigma_{ij}^m)+\frac{nm}{(n+m)^2}(\mu_i-\mu_i^m)(\mu_j-\mu_j^m). \tag{8}$$

Note that σ_{ij}^m, and therefore σ'_{ij}, can be computed in $O(m)$ time. Thus, for a fixed dimension d, the covariance matrix Σ also can be computed in $O(m)$ time.

The above derivation of the new principal components is summarized in Algorithm 3.1.

Deleting points

Let $P_m=\{\vec{p}_{n-m+1},\vec{p}_{n-m},\ldots,\vec{p}_n\}$ be a subset of the point set P, and let $\vec{\mu}^m=(\mu_1^m,\mu_2^m,\ldots,\mu_d^m)$ be the center of gravity of P_m. We subtract P_m from P, obtaining a new point set P'. The j-th component, $\mu'_j, 1\le j\le d$, of the center of gravity $\vec{\mu}'=(\mu'_1,\mu'_2,\ldots,\mu'_d)$ of P' is

$$\mu'_j=\frac{1}{n-m}\sum_{k=1}^{n-m}p_{k,j}=\frac{1}{n-m}\left(\sum_{k=1}^{n}p_{k,j}-\sum_{k=n-m+1}^{n}p_{k,j}\right)=\frac{n}{n-m}\mu_j-\frac{m}{n-m}\mu_j^m.$$

Algorithm 3.1 : ADDINGPOINTS(P, μ, Σ, P_m)

Input: point set P, the center of gravity μ of P, the covariance matrix Σ of P, point set P_m added to P

Output: principal componenets of $P \cup P_m$

1: compute the center of gravity $\vec{\mu}^m$ of P_m
2: compute the covariance matrix Σ^m of P_m
3: compute the center of gravity $\vec{\mu}'$ of $P \cup P_m$:
4: $\mu'_j = \frac{n}{n+m}\mu_j + \frac{m}{n+m}\mu_j^m$, for $1 \leq j \leq d$
5: compute the covariance Σ' of $P \cup P_m$:
6: $\sigma'_{ij} = \frac{1}{n+m}(n\sigma_{ij} + m\sigma_{ij}^m) + \frac{nm}{(n+m)^2}(\mu_i - \mu_i^m)(\mu_j - \mu_j^m)$, for $1 \leq i, j \leq d$
7: **return** the eigenvectors of Σ'

The (i, j)-th component, σ'_{ij}, $1 \leq i, j \leq d$, of the covariance matrix Σ' of P' is

$$\begin{aligned}\sigma'_{ij} &= \frac{1}{n-m}\sum_{k=1}^{n-m}(p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j) \\ &= \frac{1}{n-m}\sum_{k=1}^{n}(p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j) - \frac{1}{n-m}\sum_{k=n-m+1}^{n}(p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j).\end{aligned}$$

Let

$$\sigma'_{ij} = \sigma'_{ij,1} - \sigma'_{ij,2},$$

where

$$\sigma'_{ij,1} = \frac{1}{n-m}\sum_{k=1}^{n}(p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j), \tag{9}$$

and

$$\sigma'_{ij,2} = \frac{1}{n-m}\sum_{k=n-m+1}^{n}(p_{k,i} - \mu'_i)(p_{k,j} - \mu'_j). \tag{10}$$

Plugging-in the values of μ'_i and μ'_j in (9), we obtain:

$$\begin{aligned}\sigma'_{ij,1} &= \frac{1}{n-m}\sum_{k=1}^{n}(p_{k,i} - \frac{n}{n-m}\mu_i + \frac{m}{n-m}\mu_i^m)(p_{k,j} - \frac{n}{n-m}\mu_j + \frac{m}{n-m}\mu_j^m) \\ &= \frac{1}{n-m}\sum_{k=1}^{m}(p_{k,i} - \mu_i + \frac{m}{n-m}\mu_i - \frac{m}{n-m}\mu_i^m)(p_{k,j} - \mu_j + \frac{m}{n-m}\mu_j - \frac{m}{n-m}\mu_j^m) \\ &= \frac{1}{n-m}\sum_{k=1}^{n}(p_{k,i} - \mu_i)(p_{k,j} - \mu_j) + \frac{1}{n-m}\sum_{k=1}^{n}(p_{k,i} - \mu_i)(\frac{m}{n-m}\mu_j - \frac{m}{n-m}\mu_j^m) + \\ &\quad \frac{1}{n-m}\sum_{k=1}^{n}(\frac{m}{n-m}\mu_i - \frac{m}{n-m}\mu_i^m)(p_{k,j} - \mu_j) + \\ &\quad \frac{1}{n-m}\sum_{k=1}^{n}(\frac{m}{n-m}\mu_i - \frac{m}{n-m}\mu_i^m)(\frac{m}{n-m}\mu_j - \frac{m}{n-m}\mu_i^m).\end{aligned}$$

Since $\sum_{k=1}^{n}(p_{k,i}-\mu_i)=0, 1\le i\le d$, we have

$$\sigma'_{ij,1}=\frac{n}{n-m}\sigma_{ij}+\frac{nm^2}{(n-m)^3}(\mu_i-\mu_i^m)(\mu_j-\mu_j^m). \tag{11}$$

Plugging-in the values of μ'_i and μ'_j in (10), we obtain:

$$\begin{aligned}
\sigma'_{ij,2} &= \frac{1}{n-m}\sum_{k=n-m+1}^{n}(p_{k,i}-\frac{n}{n-m}\mu_i+\frac{m}{n-m}\mu_i^m)(p_{k,j}-\frac{n}{n-m}\mu_j+\frac{m}{n-m}\mu_j^m)\\
&= \frac{1}{n-m}\sum_{k=n-m+1}^{n}(p_{k,i}-\mu_i^m+\frac{n}{n-m}\mu_i^m-\frac{n}{n-m}\mu_i)(p_{k,j}-\mu_j^m+\frac{n}{n-m}\mu_j^m-\frac{n}{n-m}\mu_j)\\
&= \frac{1}{n-m}\sum_{k=n-m+1}^{n}(p_{k,i}-\mu_i^m)(p_{k,j}-\mu_j^m)+\frac{1}{n-m}\sum_{k=n-m+1}^{n-m}(p_{k,i}-\mu_i^m)\frac{n}{n-m}(\mu_j^m-\mu_j)+\\
&\quad \frac{1}{n-m}\sum_{k=n-m+1}^{n}\frac{n}{n-m}(\mu_i^m-\mu_i)(p_{k,j}-\mu_j^m)+\\
&\quad \frac{1}{n-m}\sum_{k=n-m+1}^{n}\frac{n}{n-m}(\mu_i^m-\mu_i)\frac{n}{n-m}(\mu_j^m-\mu_j).
\end{aligned}$$

Since $\sum_{k=n-m+1}^{n}(p_{k,i}-\mu_i^m)=0, 1\le i\le d$, we have

$$\sigma'_{ij,2}=\frac{n}{n-m}\sigma_{ij}^m+\frac{n^2m}{(n-m)^3}(\mu_i-\mu_i^m)(\mu_j-\mu_j^m), \tag{12}$$

where

$$\sigma_{ij}^m=\frac{1}{m}\sum_{k=n-m+1}^{n}(p_{k,i}-\mu_i^m)(p_{k,j}-\mu_j^m)), \quad 1\le i,j\le d,$$

is the i,j-th element of the covariance matrix Σ_m of the point set P_m.

Finally, we have

$$\sigma'_{ij}=\sigma'_{ij,1}+\sigma'_{ij,2}=\frac{1}{n-m}(n\sigma_{ij}-m\sigma_{ij}^m)-\frac{nm}{(n-m)^2}(\mu_i-\mu_i^m)(\mu_j-\mu_j^m). \tag{13}$$

Note that σ_{ij}^m, and therefore σ'_{ij}, can be computed in $O(m)$ time. Thus, for a fixed dimension d, the covariance matrix Σ also can be computed in $O(m)$ time. □

As a corollary of (13), in the case when only one point, $\vec{p}_e$, is deleted from a point set P, the elements of the new covariance matrix are given by

$$\sigma'_{ij}=\sigma'_{ij,1}-\sigma'_{ij,2}=\frac{m}{m-1}\sigma_{ij}-\frac{m}{(m-1)^2}(p_{e,i}-\mu_i)(p_{e,j}-\mu_j), \tag{14}$$

and also can be computed in $O(1)$ time.

Similar argument holds in the case when only one point is added to a point set P, and then the new covariance matrix also can be computer in constant time.

Algorithm 3.2 : DELETINGPOINTS(P, μ, Σ, P_m)

Input: point set P, the center of gravity μ of P, the covariance matrix Σ of P, point set P_m deleted from P

Output: principal components of $P \setminus P_m$

1: compute the center of gravity $\vec{\mu}^m$ of P_m
2: compute the covariance matrix Σ^m of P_m
3: compute the center of gravity $\vec{\mu}'$ of $P \setminus P_m$:
4: $\mu_j = \frac{n}{n-m}\mu_j - \frac{m}{n-m}\mu_j^m$, for $1 \leq j \leq d$
5: compute the covariance Σ' of $P \setminus P_m$:
6: $\sigma'_{ij} = \sigma'_{ij,1} + \sigma'_{ij,2} = \frac{1}{n-m}(n\sigma_{ij} - m\sigma_{ij}^m) - \frac{nm}{(n-m)^2}(\mu_i - \mu_i^m)(\mu_j - \mu_j^m)$, for $1 \leq i, j \leq d$
7: **return** the eigenvectors of Σ'

The above derivation of the new principal components is summarized in Algorithm 3.2.

Notice, that once having closed-form solutions, one can obtain similar algorithms, as presented in this chapter, when the continuous point sets are considered. Therefore, we will omit them in the next section and in the appendix.

4. Computing and updating the principal components efficiently - mboxcontinuous case $\mathbb{R}^3$

Here, we consider the computation of the principal components of a dynamic continuous point set. We present a closed form-solutions when the point set is a convex polytope or the boundary of a convex polytope in $\mathbb{R}^2$ or $\mathbb{R}^3$. When the point set is the boundary of a convex polytope, we can update the new principal components in $O(k)$ time, for both deletion and addition, under the assumption that we know the k facets in which the polytope changes. Under the same assumption, when the point set is a convex polytope in $\mathbb{R}^2$ or $\mathbb{R}^3$, we can update the principal components in $O(k)$ time after adding points. But, to update the principal components after deleting points from a convex polytope in $\mathbb{R}^2$ or $\mathbb{R}^3$ we need $O(n)$ time. This is due to the fact that, after a deletion the center of gravity of the old convex hull (polyhedron) could lie outside the new convex hull, and therefore, a retetrahedralization is needed (see Subsection 4.1 and Subsection 6.2 for details). Due to better readability and compactness of the chapter, we present in this section only the closed-form solutions for a convex polytope in $\mathbb{R}^3$, and leave the rest of the results for the appendix.

4.1 Continuous PCA over a convex polyhedron in $\mathbb{R}^3$

Let P be a point set in $\mathbb{R}^3$, and let X be its convex hull. We assume that the boundary of X is triangulated (if it is not, we can triangulate it in a preprocessing step). We choose an arbitrary point $\vec{o}$ in the interior of X, for example, we can choose $\vec{o}$ to be the center of gravity of the boundary of X. Each triangle from the boundary together with $\vec{o}$ forms a tetrahedron. Let the number of such formed tetrahedra be n. The k-th tetrahedron, with vertices $\vec{x}_{1,k}, \vec{x}_{2,k}, \vec{x}_{3,k}, \vec{x}_{4,k} = \vec{o}$, can be represented in a parametric form by $\vec{Q}_i(s,t,u) = \vec{x}_{4,i} + s\,(\vec{x}_{1,i} - \vec{x}_{4,i}) + t\,(\vec{x}_{2,i} - \vec{x}_{4,i}) + u\,(\vec{x}_{3,i} - \vec{x}_{4,i})$, for $0 \leq s,t,u \leq 1$, and $s+t+u \leq 1$. For $1 \leq i \leq 3$, we use $x_{i,j,k}$ to denote the i-th coordinate of the vertex $\vec{x}_j$ of the tetrahedron $\vec{Q}_k$.

The center of gravity of the k-th tetrahedron is

$$\vec{\mu}_k = \frac{\int_0^1 \int_0^{1-s} \int_0^{1-s-t} \rho(\vec{Q}_k(s,t))\vec{Q}_i(s,t)\, du\, dt\, ds}{\int_0^1 \int_0^{1-s} \int_0^{1-s-t} \rho(\vec{Q}_k(s,t))\, du\, dt\, ds},$$

where $\rho(\vec{Q}_k(s,t))$ is a mass density at a point $\vec{Q}_k(s,t)$. Since we can assume $\rho(\vec{Q}_k(s,t)) = 1$, we have

$$\vec{\mu}_k = \frac{\int_0^1 \int_0^{1-s} \int_0^{1-s-t} \vec{Q}_k(s,t)\, du\, dt\, ds}{\int_0^1 \int_0^{1-s} \int_0^{1-s-t} du\, dt\, ds} = \frac{\vec{x}_{1,k} + \vec{x}_{2,k} + \vec{x}_{3,k} + \vec{x}_{4,k}}{4}.$$

The contribution of each tetrahedron to the center of gravity of X is proportional to its volume. If M_k is the 3×3 matrix whose l-th row is $\vec{x}_{l,k} - \vec{x}_{4,k}$, for $l = 1 \ldots 3$, then the volume of the k-th tetrahedron is

$$v_k = \text{volume}(Q_k) = \frac{|det(M_k)|}{3!}.$$

We introduce a weight to each tetrahedron that is proportional to its volume, define as

$$w_k = \frac{v_k}{\sum_{k=1}^n v_k} = \frac{v_k}{v},$$

where v is the volume of X. Then, the center of gravity of X is

$$\vec{\mu} = \sum_{k=1}^{n} w_k \vec{\mu}_k.$$

The covariance matrix of the k-th tetrahedron is

$$\begin{aligned}\Sigma_k &= \frac{\int_0^1 \int_0^{1-s} \int_0^{1-s-t} (\vec{Q}_k(s,t,u) - \vec{\mu})\,(\vec{Q}_k(s,t,u) - \vec{\mu})^T\, du\, dt\, ds}{\int_0^1 \int_0^{1-s} \int_0^{1-s-t} du\, dt\, ds} \\ &= \frac{1}{20}\Big(\sum_{j=1}^{4} \sum_{h=1}^{4} (\vec{x}_{j,k} - \vec{\mu})(\vec{x}_{h,k} - \vec{\mu})^T + \sum_{j=1}^{4} (\vec{x}_{j,k} - \vec{\mu})(\vec{x}_{j,k} - \vec{\mu})^T \Big).\end{aligned}$$

The (i,j)-th element of Σ_k, $i,j \in \{1,2,3\}$, is

$$\sigma_{ij,k} = \frac{1}{20}\Big(\sum_{l=1}^{4} \sum_{h=1}^{4} (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \sum_{l=1}^{4} (x_{i,l,k} - \mu_i)(x_{j,l,k} - \mu_j) \Big),$$

with $\vec{\mu} = (\mu_1, \mu_2, \mu_3)$. Finally, the covariance matrix of X is

$$\Sigma = \sum_{i=1}^{n} w_i \Sigma_i,$$

with (i,j)-th element

$$\sigma_{ij} = \frac{1}{20}\Big(\sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} w_i (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \sum_{k=1}^{n} \sum_{l=1}^{4} w_i (x_{i,l,k} - \mu_i)(x_{j,l,k} - \mu_j) \Big).$$

We would like to note that the above expressions hold also for any non-convex polyhedron that can be tetrahedralized. A star-shaped object, where $\vec{o}$ is the kernel of the object, is such example.

Adding points

We add points to P, obtaining a new point set P'. Let X' be the convex hull of P'. We consider that X' is obtained from X by deleting n_d, and adding n_a tetrahedra. Let

$$v' = \sum_{k=1}^{n} v_k + \sum_{k=n+1}^{n+n_a} v_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} v_k = v + \sum_{k=n+1}^{n+n_a} v_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} v_k.$$

The center of gravity of X' is

$$\begin{aligned}\vec{\mu}' &= \sum_{k=1}^{n} w'_k\vec{\mu}_k + \sum_{k=n+1}^{n+n_a} w'_k\vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} w'_k\vec{\mu}_k \\ &= \frac{1}{v'}\left(\sum_{k=1}^{n} v_k\vec{\mu}_k + \sum_{k=n+1}^{n+n_a} v_k\vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} v_k\vec{\mu}_k\right) \\ &= \frac{1}{v'}\left(v\vec{\mu} + \sum_{k=n+1}^{n+n_a} v_k\vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} v_k\vec{\mu}_k\right). \end{aligned} \tag{15}$$

Let

$$\vec{\mu}_a = \frac{1}{v'}\sum_{k=n+1}^{n+n_a} v_k\vec{\mu}_k, \quad \text{and} \quad \vec{\mu}_d = \frac{1}{v'}\sum_{k=n+n_a+1}^{n+n_a+n_d} v_k\vec{\mu}_k.$$

Then, we can rewrite (15) as

$$\vec{\mu}' = \frac{v}{v'}\vec{\mu} + \vec{\mu}_a - \vec{\mu}_d. \tag{16}$$

The i-th component of $\vec{\mu}_a$ and $\vec{\mu}_d$, $1 \le i \le 3$, is denoted by $\mu_{i,a}$ and $\mu_{i,d}$, respectively. The (i,j)-th component, σ'_{ij}, $1 \le i,j \le 3$, of the covariance matrix Σ' of X' is

$$\begin{aligned}\sigma'_{ij} = \frac{1}{20}\Big(&\sum_{k=1}^{n}\sum_{l=1}^{4}\sum_{h=1}^{4} w'_k(x_{i,l,k}-\mu'_i)(x_{j,h,k}-\mu'_j) + \sum_{k=1}^{n}\sum_{l=1}^{4} w'_k(x_{i,l,k}-\mu'_i)(x_{j,l,k}-\mu'_j)\Big) + \\ \frac{1}{20}\Big(&\sum_{k=n+1}^{n+n_a}\sum_{l=1}^{4}\sum_{h=1}^{4} w'_k(x_{i,l,k}-\mu'_i)(x_{j,h,k}-\mu'_j) + \sum_{k=n+1}^{n+n_a}\sum_{l=1}^{4} w'_k(x_{i,l,k}-\mu'_i)(x_{j,l,k}-\mu'_j) - \\ &\sum_{k=n+n_a+1}^{n+n_a+n_d}\sum_{l=1}^{4}\sum_{h=1}^{4} w'_k(x_{i,l,k}-\mu'_i)(x_{j,h,k}-\mu'_j) - \\ &\sum_{k=n+n_a+1}^{n+n_a+n_d}\sum_{l=1}^{4} w'_k(x_{i,l,k}-\mu'_i)(x_{j,l,k}-\mu'_j)\Big).\end{aligned}$$

Let

$$\sigma'_{ij} = \frac{1}{20}(\sigma'_{ij,11} + \sigma'_{ij,12} + \sigma'_{ij,21} + \sigma'_{ij,22} - \sigma'_{ij,31} - \sigma'_{ij,32}),$$

where

$$\sigma'_{ij,11} = \sum_{k=1}^{n}\sum_{l=1}^{4}\sum_{h=1}^{4} w'_k(x_{i,l,k}-\mu'_i)(x_{j,h,k}-\mu'_j), \tag{17}$$

$$\sigma'_{ij,12} = \sum_{k=1}^{n} \sum_{l=1}^{4} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j), \tag{18}$$

$$\sigma'_{ij,21} = \sum_{k=n+1}^{n+n_a} \sum_{l=1}^{4} \sum_{h=1}^{4} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{19}$$

$$\sigma'_{ij,22} = \sum_{k=n+1}^{n+n_a} \sum_{l=1}^{4} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j), \tag{20}$$

$$\sigma'_{ij,31} = \sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{4} \sum_{h=1}^{4} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{21}$$

$$\sigma'_{ij,32} = \sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{4} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j). \tag{22}$$

Plugging-in the values of μ'_i and μ'_j in (17), we obtain:

$$\begin{aligned}
\sigma'_{ij,11} &= \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} w'_k (x_{i,l,k} - \frac{v}{v'}\mu_i - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \frac{v}{v'}\mu_j - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} w'_k (x_{i,l,k} - \mu_i + \mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d}) \cdot \\
&\qquad (x_{j,h,k} - \mu_j + \mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} w'_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} w'_k (x_{i,l,k} - \mu_i)(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} w'_k (\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} w'_k (\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}).
\end{aligned} \tag{23}$$

Since $\sum_{k=1}^{n} \sum_{l=1}^{4} w'_k (x_{i,l,k} - \mu_i) = 0, 1 \leq i \leq 3$, we have

$$\begin{aligned}
\sigma'_{ij,11} &= \frac{1}{v'} \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} v_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \frac{1}{v'} \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} v_k (\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \frac{1}{v'} \sum_{k=1}^{n} \sum_{l=1}^{4} \sum_{h=1}^{4} v_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad 16 \frac{v}{v'} (\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}).
\end{aligned} \tag{24}$$

Plugging-in the values of μ'_i and μ'_j in (18), we obtain:

$$\begin{aligned}
\sigma'_{ij,12} &= \sum_{k=1}^{n}\sum_{l=1}^{4} w'_k(x_{i,l,k} - \frac{v}{v'}\mu_i - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \frac{v}{v'}\mu_j - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n}\sum_{l=1}^{4} w'_k(x_{i,l,k} - \mu_i + \mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d}) \cdot \\
&\qquad (x_{j,h,k} - \mu_j + \mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n}\sum_{l=1}^{4} w'_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{4} w'_k(x_{i,l,k} - \mu_i)(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{4} w'_k(\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{4} w'_k(\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}). \qquad (25)
\end{aligned}$$

Since $\sum_{k=1}^{n}\sum_{l=1}^{4} w'_k(x_{i,l,k} - \mu_i) = 0$, $1 \le i \le 3$, we have

$$\begin{aligned}
\sigma'_{ij,12} &= \frac{1}{v'}\sum_{k=1}^{n}\sum_{l=1}^{4} v_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \frac{1}{v'}\sum_{k=1}^{n}\sum_{l=1}^{4} v_k(\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \frac{1}{v'}\sum_{k=1}^{n}\sum_{l=1}^{4} v_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad 4\frac{v}{v'}(\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}). \qquad (26)
\end{aligned}$$

From (25) and (26), we obtain

$$\sigma'_{ij,1} = \sigma'_{ij,11} + \sigma'_{ij,12} = \sigma_{ij} + 20\frac{v}{v'}(\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}). \qquad (27)$$

Note that $\sigma'_{ij,1}$ can be computed in $O(1)$ time. The components $\sigma'_{ij,21}$ and $\sigma'_{ij,22}$ can be computed in $O(n_a)$ time, while $O(n_d)$ time is needed to compute $\sigma'_{ij,31}$ and $\sigma'_{ij,32}$. Thus, $\vec{\mu'}$ and

$$\begin{aligned}
\sigma'_{ij} &= \frac{1}{20}(\sigma'_{ij,11} + \sigma'_{ij,12} + \sigma'_{ij,21} + \sigma'_{ij,22} + \sigma'_{ij,31} + \sigma'_{ij,32}) \\
&= \frac{1}{20}(\sigma_{ij} + \sigma'_{ij,21} + \sigma'_{ij,22} + \sigma'_{ij,31} + \sigma_{ij,32}) + \\
&\quad \frac{v}{v'}(\mu_i(1 - \frac{v}{v'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{v}{v'}) - \mu_{j,a} + \mu_{j,d}) \qquad (28)
\end{aligned}$$

can be computed in $O(n_a + n_d)$ time.

Deleting points

Let the new convex hull be obtained by deleting n_d tetrahedra from and added n_a tetrahedra to the old convex hull. If the interior point $\vec{o}$ (needed for a tetrahedronization of a convex polytope), after several deletions, lies inside the new convex hull, then the same formulas and time complexity, as by adding points, follow. If $\vec{o}$ lie outside the new convex hull, then, we need to choose a new interior point $\vec{o}'$, and recompute the new tetrahedra associated with it. Thus, we need in total $O(n)$ time to update the principal components.

Under certain assumptions, we can recompute the new principal components faster:

- If we know that a certain point of the polyhedron will never be deleted, we can choose $\vec{o}$ to be that point. In that case, we also have the same closed-formed solution as for adding a point.
- Let the facets of the convex polyhedron have similar (uniformly distributed) area. We choose $\vec{o}$ to be the center of gravity of the polyhedron. Then, we can expect that after deleting a point, $\vec{o}$ will remain in the new convex hull. However, after several deletions, $\vec{o}$ could lie outside the convex hull, and then we need to recompute it and the tetrahedra associated with it.

Note that in the case when we consider boundary of a convex polyhedron (Subsection 6.1 and Subsection 6.3), we do not need an interior point $\vec{o}$ and the same time complexity holds for both adding and deleting points.

5. Conclusion

In this chapter, we have presented closed-form solutions for computing and updating the principal components of (dynamic) discrete and continuous point sets. The new principal components can be computed in constant time, when a constant number of points are added or deleted from the point set. This is a significant improvement of the commonly used approach, when the new principal components are computed from scratch, which takes linear time. The advantages of some of the theoretical results were verified and presented in the context of computing dynamic PCA bounding boxes in Dimitrov, Holst, Knauer & Kriegel (2009); Dimitrov et al. (2011).

An interesting open problem is to find a closed-form solutions for dynamical point sets different from convex polyhedra, for example, implicit surfaces or B-splines. An implementation of computing principal components in a dynamic, continuous setting could be a useful practical extension of the results presented here regarding continuous point sets. Applications of the results presented here in other fields, like computer vision or visualization, are of high interest.

6. Appendix: Computing and updating the principal components efficiently - continuous case

6.1 Continuous PCA over the boundary of a polyhedron in $\mathbb{R}^3$.

Let X be a polyhedron in $\mathbb{R}^3$. We assume that the boundary of X is triangulated (if it is not, we can triangulate it in a preprocessing), containing n triangles. The k-th triangle, with vertices $\vec{x}_{1,k}, \vec{x}_{2,k}, \vec{x}_{3,k}$, can be represented in a parametric form by $\vec{T}_k(s,t) = \vec{x}_{1,k} + s\,(\vec{x}_{2,k} - \vec{x}_{1,k}) +$

$t\,(\vec{x}_{3,k} - \vec{x}_{1,k})$, for $0 \leq s,t \leq 1$, and $s+t \leq 1$. For $1 \leq i \leq 3$, we denote by $x_{i,j,k}$ the i-th coordinate of the vertex $\vec{x}_j$ of the triangle $\vec{T}_k$. The center of gravity of the k-th triangle is

$$\vec{\mu}_k = \frac{\int_0^1 \int_0^{1-s} \vec{T}_i(s,t)\, dt\, ds}{\int_0^1 \int_0^{1-s} dt\, ds} = \frac{\vec{x}_{1,k} + \vec{x}_{2,k} + \vec{x}_{3,k}}{3}.$$

The contribution of each triangle to the center of gravity of the triangulated surface is proportional to its area. The area of the k-th triangle is

$$a_k = \text{area}(T_k) = \frac{|(\vec{x}_{2,k} - \vec{x}_{1,k})| \times |(\vec{x}_{3,k} - \vec{x}_{1,k})|}{2}.$$

We introduce a weight to each triangle that is proportional to its area, define as

$$w_k = \frac{a_k}{\sum_{i=1}^n a_k} = \frac{a_k}{a},$$

where a is the area of X. Then, the center of gravity of the boundary of X is

$$\vec{\mu} = \sum_{k=1}^{n} w_k \vec{\mu}_k.$$

The covariance matrix of the k-th triangle is

$$\begin{aligned}\Sigma_k &= \frac{\int_0^1 \int_0^{1-s} (\vec{T}_k(s,t) - \vec{\mu})\,(\vec{T}_k(s,t) - \vec{\mu})^T\, dt\, ds}{\int_0^1 \int_0^{1-s} dt\, ds} \\ &= \frac{1}{12}\Big(\sum_{j=1}^{3} \sum_{h=1}^{3} (\vec{x}_{j,k} - \vec{\mu})(\vec{x}_{h,k} - \vec{\mu})^T + \sum_{j=1}^{3} (\vec{x}_{j,k} - \vec{\mu})(\vec{x}_{j,k} - \vec{\mu})^T \Big).\end{aligned}$$

The (i,j)-th element of Σ_k, $i,j \in \{1,2,3\}$, is

$$\sigma_{ij,k} = \frac{1}{12}\Big(\sum_{l=1}^{3} \sum_{h=1}^{3} (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \sum_{l=1}^{3} (x_{i,l,k} - \mu_i)(x_{j,l,k} - \mu_j) \Big),$$

with $\vec{\mu} = (\mu_1, \mu_2, \mu_3)$. Finally, the covariance matrix of the boundary of X is

$$\Sigma = \sum_{k=1}^{n} w_k \Sigma_k.$$

Adding points

We add points to X. Let X' be the new convex hull. We assume that X' is obtained from X by deleting n_d, and adding n_a tetrahedra. Then the sum of the areas of all triangles is

$$a' = \sum_{k=1}^{n} a_k + \sum_{k=n+1}^{n+n_a} a_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k = a + \sum_{k=n+1}^{n+n_a} a_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k.$$

The center of gravity of X' is

$$\vec{\mu}' = \sum_{k=1}^{n} w'_k \vec{\mu}_k + \sum_{k=n+1}^{n+n_a} w'_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} w'_k \vec{\mu}_k = \frac{1}{a'} \left(\sum_{k=1}^{n} a_k \vec{\mu}_k + \sum_{k=n+1}^{n+n_a} a_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k \vec{\mu}_k \right)$$
$$= \frac{1}{a'} \left(a\vec{\mu} + \sum_{k=n+1}^{n+n_a} a_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k \vec{\mu}_k \right). \quad (29)$$

Let

$$\vec{\mu}_a = \frac{1}{a'} \sum_{k=n+1}^{n+n_a} a_k \vec{\mu}_k, \quad \text{and} \quad \vec{\mu}_d = \frac{1}{a'} \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k \vec{\mu}_k.$$

Then, we can rewrite (29) as

$$\vec{\mu}' = \frac{a}{a'} \vec{\mu} + \vec{\mu}_a - \vec{\mu}_d. \quad (30)$$

The i-th component of $\vec{\mu}_a$ and $\vec{\mu}_d$, $1 \leq i \leq 3$, is denoted by $\mu_{i,a}$ and $\mu_{i,d}$, respectively. The (i,j)-th component, σ'_{ij}, $1 \leq i,j \leq 3$, of the covariance matrix Σ' of X' is

$$\sigma'_{ij} = \frac{1}{12} \Big(\sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) + \sum_{k=1}^{n} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j) \Big) +$$
$$\frac{1}{12} \Big(\sum_{k=n+1}^{n+n_a} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) + \sum_{k=n+1}^{n+n_a} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j) -$$
$$\sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) -$$
$$\sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j) \Big).$$

Let

$$\sigma'_{ij} = \frac{1}{12} (\sigma'_{ij,11} + \sigma'_{ij,12} + \sigma'_{ij,21} + \sigma'_{ij,22} - \sigma'_{ij,31} - \sigma'_{ij,32}),$$

where

$$\sigma'_{ij,11} = \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \quad (31)$$

$$\sigma'_{ij,12} = \sum_{k=1}^{n} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j), \quad (32)$$

$$1.5mm\sigma'_{ij,21} = \sum_{k=n+1}^{n+n_a} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \quad (33)$$

$$1.5mm\sigma'_{ij,22} = \sum_{k=n+1}^{n+n_a} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j), \quad (34)$$

$$\sigma'_{ij,31} = \sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{35}$$

$$\sigma'_{ij,32} = \sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j). \tag{36}$$

Plugging-in the values of μ'_i and μ'_j in (31), we obtain:

$$\begin{aligned}
\sigma'_{ij,11} &= \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \frac{a}{a'}\mu_i - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \frac{a}{a'}\mu_j - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu_i + \mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d}) \cdot \\
&\qquad (x_{j,h,k} - \mu_j + \mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu_i)(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}).
\end{aligned} \tag{37}$$

Since $\sum_{k=1}^{n} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu_i) = 0$, $1 \leq i \leq 3$, we have

$$\begin{aligned}
\sigma'_{ij,11} &= \frac{1}{a'} \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} a_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \frac{1}{a'} \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} a_k (\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \frac{1}{a'} \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} a_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad 9\frac{a}{a'}(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}).
\end{aligned} \tag{38}$$

Plugging-in the values of μ'_i and μ'_j in (32), we obtain:

$$\begin{aligned}
\sigma'_{ij,12} &= \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \frac{a}{a'}\mu_i - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \frac{a}{a'}\mu_j - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu_i + \mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d}) \cdot \\
&\qquad (x_{j,h,k} - \mu_j + \mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu_i)(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}). \qquad (39)
\end{aligned}$$

Since $\sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu_i) = 0$, $1 \le i \le 3$, we have

$$\begin{aligned}
\sigma'_{ij,12} &= \frac{1}{a'}\sum_{k=1}^{n}\sum_{l=1}^{3} a_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \frac{1}{a'}\sum_{k=1}^{n}\sum_{l=1}^{3} a_k(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \frac{1}{a'}\sum_{k=1}^{n}\sum_{l=1}^{3} a_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad 3\frac{a}{a'}(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}). \qquad (40)
\end{aligned}$$

From (39) and (40), we obtain

$$\begin{aligned}
\sigma'_{ij,1} &= \sigma'_{ij,11} + \sigma'_{ij,12} \\
&= \sigma_{ij} + 12\frac{a}{a'}(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}). \qquad (41)
\end{aligned}$$

Note that $\sigma'_{ij,1}$ can be computed in $O(1)$ time. The components $\sigma'_{ij,21}$ and $\sigma'_{ij,22}$ can be computed in $O(n_a)$ time, while $O(n_d)$ time is needed to compute $\sigma'_{ij,31}$ and $\sigma'_{ij,32}$. Thus, $\vec{\mu'}$ and

$$\begin{aligned}
\sigma'_{ij} &= \frac{1}{12}(\sigma'_{ij,11} + \sigma'_{ij,12} + \sigma'_{ij,21} + \sigma'_{ij,22} + \sigma'_{ij,31} + \sigma'_{ij,32}) = \frac{1}{12}(\sigma_{ij} + \sigma'_{ij,21} + \sigma'_{ij,22} + \sigma'_{ij,31} + \sigma_{ij,32}) + \\
&\quad \frac{a}{a'}(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}). \qquad (42)
\end{aligned}$$

can be computed in $O(n_a + n_d)$ time.

Deleting points

Let the new convex hull be obtained by deleting n_d tetrahedra from and added n_a tetrahedra to the old convex hull. Consequently, the same formulas and time complexity, as by adding points, follow.

6.2 Continuous PCA over a polygon in $\mathbb{R}^2$

We assume that the polygon X is triangulated (if it is not, we can triangulate it in a preprocessing), and the number of triangles is n. The k-th triangle, with vertices $\vec{x}_{1,k}, \vec{x}_{2,k}, \vec{x}_{3,k} = \vec{o}$, can be represented in a parametric form by $\vec{T}_i(s,t) = \vec{x}_{3,k} + s\,(\vec{x}_{1,k} - \vec{x}_{3,k}) + t\,(\vec{x}_{2,k} - \vec{x}_{3,k})$, for $0 \leq s, t \leq 1$, and $s + t \leq 1$. The center of gravity of the k-th triangle is

$$\vec{\mu}_i = \frac{\int_0^1 \int_0^{1-s} \vec{T}_i(s,t)\, dt\, ds}{\int_0^1 \int_0^{1-s} dt\, ds} = \frac{\vec{x}_{1,k} + \vec{x}_{2,k} + \vec{x}_{3,k}}{3}.$$

The contribution of each triangle to the center of gravity of X is proportional to its area. The area of the i-th triangle is

$$a_k = \text{area}(T_k) = \frac{|(\vec{x}_{2,k} - \vec{x}_{1,k})| \times |(\vec{x}_{3,k} - \vec{x}_{1,k})|}{2},$$

where $\times$ denotes the vector product. We introduce a weight to each triangle that is proportional to its area, define as

$$w_k = \frac{a_k}{\sum_{k=1}^{n} a_k} = \frac{a_k}{a},$$

where a is the area of X.Then, the center of gravity of X is

$$\vec{\mu} = \sum_{k=1}^{n} w_k \vec{\mu}_k.$$

The covariance matrix of the k-th triangle is

$$\begin{aligned}\Sigma_k &= \frac{\int_0^1 \int_0^{1-s} (\vec{T}_k(s,t) - \vec{\mu})\,(\vec{T}_k(s,t) - \vec{\mu})^T\, dt\, ds}{\int_0^1 \int_0^{1-s} dt\, ds} \\ &= \frac{1}{12}\Big(\sum_{j=1}^{3} \sum_{h=1}^{3} (\vec{x}_{j,k} - \vec{\mu})(\vec{x}_{h,k} - \vec{\mu})^T + \sum_{j=1}^{3} (\vec{x}_{j,k} - \vec{\mu})(\vec{x}_{j,k} - \vec{\mu})^T \Big).\end{aligned}$$

The (i,j)-th element of Σ_k, $i, j \in \{1,2\}$, is

$$\sigma_{ij,k} = \frac{1}{12}\Big(\sum_{l=1}^{3} \sum_{h=1}^{3} (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \sum_{l=1}^{3} (x_{i,l,k} - \mu_i)(x_{j,l,k} - \mu_j) \Big),$$

with $\vec{\mu} = (\mu_1, \mu_2)$. The covariance matrix of X is

$$\Sigma = \sum_{k=1}^{n} w_k \Sigma_k.$$

Adding points

We add points to X. Let X' be the new convex hull. We assume that X' is obtained from X by deleting n_d, and adding n_a triangles. Then the sum of the areas of all triangles is

$$a' = \sum_{k=1}^{n} a_k + \sum_{k=n+1}^{n+n_a} a_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k = a + \sum_{k=n+1}^{n+n_a} a_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k.$$

The center of gravity of X' is

$$\begin{aligned}\vec{\mu}' &= \sum_{k=1}^{n} w'_k \vec{\mu}_k + \sum_{k=n+1}^{n+n_a} w'_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} w'_k \vec{\mu}_k = \frac{1}{a'}\left(\sum_{k=1}^{n} a_k \vec{\mu}_k + \sum_{k=n+1}^{n+n_a} a_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k \vec{\mu}_k\right) \\ &= \frac{1}{a'}\left(a\vec{\mu} + \sum_{k=n+1}^{n+n_a} a_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k \vec{\mu}_k\right). \end{aligned} \tag{43}$$

Let

$$\vec{\mu}_a = \frac{1}{a'} \sum_{k=n+1}^{n+n_a} a_k \vec{\mu}_k, \quad \text{and} \quad \vec{\mu}_d = \frac{1}{a'} \sum_{k=n+n_a+1}^{n+n_a+n_d} a_k \vec{\mu}_k.$$

Then, we can rewrite (43) as

$$\vec{\mu}' = \frac{a}{a'}\vec{\mu} + \vec{\mu}_a - \vec{\mu}_d. \tag{44}$$

The i-th component of $\vec{\mu}_a$ and $\vec{\mu}_d$, $1 \leq i \leq 2$, is denoted by $\mu_{i,a}$ and $\mu_{i,d}$, respectively. The (i,j)-th component, σ'_{ij}, $1 \leq i,j \leq 2$, of the covariance matrix Σ' of X' is

$$\begin{aligned}\sigma'_{ij} = \frac{1}{12}\Big(&\sum_{k=1}^{n}\sum_{l=1}^{3}\sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) + \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j)\Big) + \\ \frac{1}{12}\Big(&\sum_{k=n+1}^{n+n_a}\sum_{l=1}^{3}\sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) + \sum_{k=n+1}^{n+n_a}\sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j) - \\ &\sum_{k=n+n_a+1}^{n+n_a+n_d}\sum_{l=1}^{3}\sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) - \\ &\sum_{k=n+n_a+1}^{n+n_a+n_d}\sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j)\Big).\end{aligned}$$

Let

$$\sigma'_{ij} = \frac{1}{12}(\sigma'_{ij,11} + \sigma'_{ij,12} + \sigma'_{ij,21} + \sigma'_{ij,22} - \sigma'_{ij,31} - \sigma'_{ij,32}),$$

where

$$\sigma'_{ij,11} = \sum_{k=1}^{n}\sum_{l=1}^{3}\sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{45}$$

$$\sigma'_{ij,12} = \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j), \tag{46}$$

$$\sigma'_{ij,21} = \sum_{k=n+1}^{n+n_a}\sum_{l=1}^{3}\sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{47}$$

$$\sigma'_{ij,22} = \sum_{k=n+1}^{n+n_a} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j), \tag{48}$$

$$\sigma'_{ij,31} = \sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{49}$$

$$\sigma'_{ij,32} = \sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j). \tag{50}$$

Plugging-in the values of μ'_i and μ'_j in (45), we obtain:

$$\begin{aligned}
\sigma'_{ij,11} &= \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \frac{a}{a'}\mu_i - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \frac{a}{a'}\mu_j - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu_i + \mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d}) \cdot \\
&\qquad (x_{j,h,k} - \mu_j + \mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (x_{i,l,k} - \mu_i)(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} w'_k (\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}).
\end{aligned} \tag{51}$$

Since $\sum_{k=1}^{n} \sum_{l=1}^{3} w'_k (x_{i,l,k} - \mu_i) = 0$, $1 \leq i \leq 2$, we have

$$\begin{aligned}
\sigma'_{ij,11} &= \frac{1}{a'} \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} a_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \frac{1}{a'} \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} a_k (\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \frac{1}{a'} \sum_{k=1}^{n} \sum_{l=1}^{3} \sum_{h=1}^{3} a_k (x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad 9\frac{a}{a'}(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}).
\end{aligned} \tag{52}$$

Plugging-in the values of μ'_i and μ'_j in (46), we obtain:

$$\begin{aligned}
\sigma'_{ij,12} &= \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \frac{a}{a'}\mu_i - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \frac{a}{a'}\mu_j - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu_i + \mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d}) \cdot \\
&\qquad (x_{j,h,k} - \mu_j + \mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu_i)(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}). \qquad (53)
\end{aligned}$$

Since $\sum_{k=1}^{n}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu_i) = 0$, $1 \le i \le 2$, we have

$$\begin{aligned}
\sigma'_{ij,12} &= \frac{1}{a'}\sum_{k=1}^{n}\sum_{l=1}^{3} a_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \frac{1}{a'}\sum_{k=1}^{n}\sum_{l=1}^{3} a_k(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \frac{1}{a'}\sum_{k=1}^{n}\sum_{l=1}^{3} a_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad 3\frac{a}{a'}(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}). \qquad (54)
\end{aligned}$$

From (53) and (54), we obtain

$$\sigma'_{ij,1} = \sigma'_{ij,11} + \sigma'_{ij,12} = \sigma_{ij} + 12\frac{a}{a'}(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}). \qquad (55)$$

Note that $\sigma'_{ij,1}$ can be computed in $O(1)$ time. The components $\sigma'_{ij,21}$ and $\sigma'_{ij,22}$ can be computed in $O(n_a)$ time, while $O(n_d)$ time is needed to compute $\sigma'_{ij,31}$ and $\sigma'_{ij,32}$. Thus, $\vec{\mu'}$ and

$$\begin{aligned}
\sigma'_{ij} &= \frac{1}{12}(\sigma'_{ij,11} + \sigma'_{ij,12} + \sigma'_{ij,21} + \sigma'_{ij,22} + \sigma'_{ij,31} + \sigma'_{ij,32}) \\
&= \frac{1}{12}(\sigma_{ij} + \sigma'_{ij,21} + \sigma'_{ij,22} + \sigma'_{ij,31} + \sigma_{ij,32}) + \\
&\quad \frac{a}{a'}(\mu_i(1 - \frac{a}{a'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{a}{a'}) - \mu_{j,a} + \mu_{j,d}). \qquad (56)
\end{aligned}$$

can be computed in $O(n_a + n_d)$ time.

Deleting points

Let the new convex hull be obtained by deleting n_d tetrahedra from and added n_a tetrahedra to the old convex hull. If the interior point $\vec{o}$, after deleting points, lies inside the new convex hull, then the same formulas and time complexity, as by adding points, follow. However, $\vec{o}$ could lie outside the new convex hull. Then, we need to choose a new interior point $\vec{o}'$, and recompute the new tetrahedra associated with it. Thus, we need in total $O(n)$ time to update the principal components.

6.3 Continuous PCA over the boundary of a polygon $\mathbb{R}^2$

Let X be a polygon in $\mathbb{R}^2$. We assume that the boundary of X is comprised of n line segments. The k-th line segment, with vertices $\vec{x}_{1,k}, \vec{x}_{2,k}$, can be represented in a parametric form by

$$\vec{S}_k(t) = \vec{x}_{1,k} + t\,(\vec{x}_{2,k} - \vec{x}_{1,k}).$$

Since we assume that the mass density is constant, the center of gravity of the k-th line segment is

$$\vec{\mu}_k = \frac{\int_0^1 \vec{S}_k(t)\,dt}{\int_0^1 dt} = \frac{\vec{x}_{1,k} + \vec{x}_{2,k}}{2}.$$

The contribution of each line segment to the center of gravity of the boundary of a polygon is proportional to the length of the line segment. The length of the k-th line segment is

$$s_k = \text{length}(S_k) = ||\vec{x}_{2,k} - \vec{x}_{1,k}||.$$

We introduce a weight to each line segment that is proportional to its length, define as

$$w_k = \frac{s_k}{\sum_{k=1}^n s_k} = \frac{s_k}{s},$$

where s is the perimeter of X. Then, the center of gravity of the boundary of X is

$$\vec{\mu} = \sum_{k=1}^{n} w_k \vec{\mu}_k.$$

The covariance matrix of the k-th line segment is

$$\begin{aligned}\Sigma_k &= \frac{\int_0^1 (\vec{S}_k(t) - \vec{\mu})\,(\vec{S}_k(t) - \vec{\mu})^T\,dt}{\int_0^1 dt} \\ &= \frac{1}{6}\Big(\sum_{j=1}^{2}\sum_{h=1}^{2}(\vec{x}_{j,k} - \vec{\mu})(\vec{x}_{h,k} - \vec{\mu})^T + \sum_{j=1}^{2}(\vec{x}_{j,k} - \vec{\mu})(\vec{x}_{j,k} - \vec{\mu})^T\Big).\end{aligned}$$

The (i,j)-th element of Σ_k, $i,j \in \{1,2\}$, is

$$\sigma_{ij,k} = \frac{1}{6}\Big(\sum_{l=1}^{2}\sum_{h=1}^{2}(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \sum_{l=1}^{2}(x_{i,l,k} - \mu_i)(x_{j,l,k} - \mu_j)\Big),$$

with $\vec{\mu} = (\mu_1, \mu_2)$.

The covariance matrix of the boundary of X is

$$\Sigma = \sum_{k=1}^{n} w_k \Sigma_k.$$

Adding points

We add points to X. Let X' be the new convex hull. We assume that X' is obtained from X by deleting n_d, and adding n_a line segments. Then the sum of the lengths of all line segments is

$$s' = \sum_{k=1}^{n} l_k + \sum_{k=n+1}^{n+n_a} s_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} s_k = s + \sum_{k=n+1}^{n+n_a} s_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} s_k.$$

The center of gravity of X' is

$$\begin{aligned}
\vec{\mu}' &= \sum_{k=1}^{n} w'_k \vec{\mu}_k + \sum_{k=n+1}^{n+n_a} w'_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} w'_k \vec{\mu}_k \\
&= \frac{1}{s'} \left(\sum_{k=1}^{n} s_k \vec{\mu}_k + \sum_{k=n+1}^{n+n_a} s_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} s_k \vec{\mu}_k \right) \\
&= \frac{1}{s'} \left(s\vec{\mu} + \sum_{k=n+1}^{n+n_a} s_k \vec{\mu}_k - \sum_{k=n+n_a+1}^{n+n_a+n_d} s_k \vec{\mu}_k \right). \qquad (57)
\end{aligned}$$

Let

$$\vec{\mu}_a = \frac{1}{s'} \sum_{k=n+1}^{n+n_a} s_k \vec{\mu}_k, \quad \text{and} \quad \vec{\mu}_d = \frac{1}{s'} \sum_{k=n+n_a+1}^{n+n_a+n_d} s_k \vec{\mu}_k.$$

Then, we can rewrite (57) as

$$\vec{\mu}' = \frac{s}{s'} \vec{\mu} + \vec{\mu}_a - \vec{\mu}_d. \qquad (58)$$

The i-th component of $\vec{\mu}_a$ and $\vec{\mu}_d$, $1 \leq i \leq 2$, is denoted by $\mu_{i,a}$ and $\mu_{i,d}$, respectively. The (i,j)-th component, σ'_{ij}, $1 \leq i,j \leq 2$, of the covariance matrix Σ' of X' is

$$\begin{aligned}
\sigma'_{ij} = \frac{1}{6} \Big(& \sum_{k=1}^{n} \sum_{l=1}^{2} \sum_{h=1}^{2} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) + \sum_{k=1}^{n} \sum_{l=1}^{2} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j) \Big) + \\
& \frac{1}{6} \Big(\sum_{k=n+1}^{n+n_a} \sum_{l=1}^{2} \sum_{h=1}^{2} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) + \sum_{k=n+1}^{n+n_a} \sum_{l=1}^{2} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j) - \\
& \sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{2} \sum_{h=1}^{2} w'_k (x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j) - \\
& \sum_{k=n+n_a+1}^{n+n_a+n_d} \sum_{l=1}^{2} w'_k (x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j) \Big).
\end{aligned}$$

Let

$$\sigma'_{ij} = \frac{1}{6}(\sigma'_{ij,11} + \sigma'_{ij,12} + \sigma'_{ij,21} + \sigma'_{ij,22} - \sigma'_{ij,31} - \sigma'_{ij,32}),$$

where

$$\sigma'_{ij,11} = \sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{59}$$

$$\sigma'_{ij,12} = \sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j), \tag{60}$$

$$\sigma'_{ij,21} = \sum_{k=n+1}^{n+n_a}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{61}$$

$$\sigma'_{ij,22} = \sum_{k=n+1}^{n+n_a}\sum_{l=1}^{2} w'_k(x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j), \tag{62}$$

$$\sigma'_{ij,31} = \sum_{k=n+n_a+1}^{n+n_a+n_d}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(x_{i,l,k} - \mu'_i)(x_{j,h,k} - \mu'_j), \tag{63}$$

$$\sigma'_{ij,32} = \sum_{k=n+n_a+1}^{n+n_a+n_d}\sum_{l=1}^{3} w'_k(x_{i,l,k} - \mu'_i)(x_{j,l,k} - \mu'_j). \tag{64}$$

Plugging-in the values of μ'_i and μ'_j in (59), we obtain:

$$\begin{aligned}
\sigma'_{ij,11} &= \sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(x_{i,l,k} - \frac{s}{s'}\mu_i - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \frac{s}{s'}\mu_j - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(x_{i,l,k} - \mu_i + \mu_i(1 - \frac{s}{s'}) - \mu_{i,a} + \mu_{i,d}) \cdot \\
&\qquad (x_{j,h,k} - \mu_j + \mu_j(1 - \frac{s}{s'}) - \mu_{j,a} + \mu_{j,d}) \\
&= \sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(x_{i,l,k} - \mu_i)(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(x_{i,l,k} - \mu_i)(\mu_j(1 - \frac{s}{s'}) - \mu_{j,a} + \mu_{j,d}) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(\mu_i(1 - \frac{s}{s'}) - \mu_{i,a} + \mu_{i,d})(x_{j,h,k} - \mu_j) + \\
&\quad \sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} w'_k(\mu_i(1 - \frac{s}{s'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1 - \frac{s}{s'}) - \mu_{j,a} + \mu_{j,d}).
\end{aligned} \tag{65}$$

Since $\sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(x_{i,l,k}-\mu_i)=0, 1\leq i\leq 2$, we have

$$\begin{aligned}\sigma'_{ij,11} &= \frac{1}{s'}\sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} s_k(x_{i,l,k}-\mu_i)(x_{j,h,k}-\mu_j)+\\ &\quad \frac{1}{s'}\sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} s_k(\mu_i(1-\frac{s}{s'})-\mu_{i,a}+\mu_{i,d})(\mu_j(1-\frac{s}{s'})-\mu_{j,a}+\mu_{j,d})\\ &= \frac{1}{s'}\sum_{k=1}^{n}\sum_{l=1}^{2}\sum_{h=1}^{2} s_k(x_{i,l,k}-\mu_i)(x_{j,h,k}-\mu_j)+\\ &\quad 4\frac{s}{s'}(\mu_i(1-\frac{s}{s'})-\mu_{i,a}+\mu_{i,d})(\mu_j(1-\frac{s}{s'})-\mu_{j,a}+\mu_{j,d}). \end{aligned} \tag{66}$$

Plugging-in the values of μ'_i and μ'_j in (60), we obtain:

$$\begin{aligned}\sigma'_{ij,12} &= \sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(x_{i,l,k}-\frac{s}{s'}\mu_i-\mu_{i,a}+\mu_{i,d})(x_{j,h,k}-\frac{s}{s'}\mu_j-\mu_{j,a}+\mu_{j,d})\\ &= \sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(x_{i,l,k}-\mu_i+\mu_i(1-\frac{s}{s'})-\mu_{i,a}+\mu_{i,d})\cdot\\ &\qquad (x_{j,h,k}-\mu_j+\mu_j(1-\frac{s}{s'})-\mu_{j,a}+\mu_{j,d})\\ &= \sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(x_{i,l,k}-\mu_i)(x_{j,h,k}-\mu_j)+\\ &\quad \sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(x_{i,l,k}-\mu_i)(\mu_j(1-\frac{s}{s'})-\mu_{j,a}+\mu_{j,d})+\\ &\quad \sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(\mu_i(1-\frac{s}{s'})-\mu_{i,a}+\mu_{i,d})(x_{j,h,k}-\mu_j)+\\ &\quad \sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(\mu_i(1-\frac{s}{s'})-\mu_{i,a}+\mu_{i,d})(\mu_j(1-\frac{s}{s'})-\mu_{j,a}+\mu_{j,d}). \end{aligned} \tag{67}$$

Since $\sum_{k=1}^{n}\sum_{l=1}^{2} w'_k(x_{i,l,k}-\mu_i)=0, 1\leq i\leq 2$, we have

$$\begin{aligned}\sigma'_{ij,12} &= \frac{1}{s'}\sum_{k=1}^{n}\sum_{l=1}^{2} s_k(x_{i,l,k}-\mu_i)(x_{j,h,k}-\mu_j)+\\ &\quad \frac{1}{s'}\sum_{k=1}^{n}\sum_{l=1}^{2} s_k(\mu_i(1-\frac{s}{s'})-\mu_{i,a}+\mu_{i,d})(\mu_j(1-\frac{s}{s'})-\mu_{j,a}+\mu_{j,d})\\ &= \frac{1}{s'}\sum_{k=1}^{n}\sum_{l=1}^{2} s_k(x_{i,l,k}-\mu_i)(x_{j,h,k}-\mu_j)+\\ &\quad 2\frac{s}{s'}(\mu_i(1-\frac{s}{s'})-\mu_{i,a}+\mu_{i,d})(\mu_j(1-\frac{s}{s'})-\mu_{j,a}+\mu_{j,d}). \end{aligned} \tag{68}$$

From (67) and (68), we obtain

$$\begin{aligned}\sigma'_{ij,1} &= \sigma'_{ij,11} + \sigma'_{ij,12} \\ &= \sigma_{ij} + 6\frac{s}{s'}(\mu_i(1-\frac{s}{s'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1-\frac{s}{s'}) - \mu_{j,a} + \mu_{j,d}). \end{aligned} \tag{69}$$

Note that $\sigma'_{ij,1}$ can be computed in $O(1)$ time. The components $\sigma'_{ij,21}$ and $\sigma'_{ij,22}$ can be computed in $O(n_a)$ time, while $O(n_d)$ time is needed to compute $\sigma'_{ij,31}$ and $\sigma'_{ij,32}$. Thus, $\vec{\mu'}$ and

$$\begin{aligned}\sigma'_{ij} &= \frac{1}{6}(\sigma'_{ij,11} + \sigma'_{ij,12} + \sigma'_{ij,21} + \sigma'_{ij,22} + \sigma'_{ij,31} + \sigma'_{ij,32}) \\ &= \frac{1}{6}(\sigma_{ij} + \sigma'_{ij,21} + \sigma'_{ij,22} + \sigma'_{ij,31} + \sigma_{ij,32}) + \\ &\quad \frac{s}{s'}(\mu_i(1-\frac{s}{s'}) - \mu_{i,a} + \mu_{i,d})(\mu_j(1-\frac{s}{s'}) - \mu_{j,a} + \mu_{j,d}) \end{aligned} \tag{70}$$

can be computed in $O(n_a + n_d)$ time.

Deleting points

Let the new convex hull be obtained by deleting n_d tetrahedra from and added n_a tetrahedra to the old convex hull. Consequently, the same formulas and time complexity, as by adding points, follow.

7. References

Chan, T. F., Golub, G. H. & LeVeque, R. J. (1979). Updating formulae and a pairwise algorithm for computing sample variances, *Technical Report STAN-CS-79-773*, Department of Computer Science, Stanford University.

Cheng, S.-W. & Y. Wang, Z. W. (2008). Provable dimension detection using principal component analysis, *Int. J. Comput. Geometry Appl.* 18: 415–440.

Dimitrov, D., Holst, M., Knauer, C. & Kriegel, K. (2009). Closed-form solutions for continuous PCA and bounding box algorithms, *A. Ranchordas et al. (Eds.): VISIGRAPP 2008, CCIS, Springer* 24: 26–40.

Dimitrov, D., Holst, M., Knauer, C. & Kriegel, K. (2011). Efficient dynamical computation of principal components, *Proceedings of International Conference on Computer Graphics Theory and Applications - GRAPP*, pp. 85–93.

Dimitrov, D., Knauer, C., Kriegel, K. & Rote, G. (2009). Bounds on the quality of the PCA bounding boxes, *Computational Geometry* 42: 772–789.

Duda, R., Hart, P. & Stork, D. (2001). *Pattern classification*, John Wiley & Sons, Inc., 2nd ed.

Gottschalk, S., Lin, M. C. & Manocha, D. (1996). OBBTree: A hierarchical structure for rapid interference detection, *Computer Graphics* 30: 171–180.

Jolliffe, I. (2002). *Principal Component Analysis*, Springer-Verlag, New York, 2nd ed.

Knuth, D. E. (1998). *The art of computer programming, volume 2: seminumerical algorithms*, Addison-Wesley, Boston, 3rd ed.

Parlett, B. N. (1998). *The symmetric eigenvalue problem*, Society of Industrial and Applied Mathematics (SIAM), Philadelphia, PA.

Pébay, P. P. (2008). Formulas for robust, one-pass parallel computation of covariances and arbitrary-order statistical moments, *Technical Report SAND2008-6212*, Sandia National Laboratories.

Press, W. H., Teukolsky, S. A., Veterling, W. T. & Flannery, B. P. (1995). *Numerical recipes in C: the art of scientific computing*, Cambridge University Press, New York, USA, 2nd ed.

Vranić, D. V., Saupe, D. & Richter, J. (2001). Tools for 3D-object retrieval: Karhunen-Loeve transform and spherical harmonics, *IEEE 2001 Workshop Multimedia Signal Processing*, pp. 293–298.

Welford, B. P. (1962). Note on a method for calculating corrected sums of squares and products, *Technometrics* 4: 419–420.

West, D. H. D. (1979). Updating mean and variance estimates: an improved method, *Communications of the ACM* 22: 532–535.

IntechOpen



Principal Component Analysis

Edited by Dr. Parinya Sanguansat

ISBN 978-953-51-0195-6
Hard cover, 300 pages
Publisher InTech
Published online 02, March, 2012
Published in print edition March, 2012

This book is aimed at raising awareness of researchers, scientists and engineers on the benefits of Principal Component Analysis (PCA) in data analysis. In this book, the reader will find the applications of PCA in fields such as image processing, biometric, face recognition and speech processing. It also includes the core concepts and the state-of-the-art methods in data analysis and feature extraction.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Darko Dimitrov (2012). Computing and Updating Principal Components of Discrete and Continuous Point Sets, Principal Component Analysis, Dr. Parinya Sanguansat (Ed.), ISBN: 978-953-51-0195-6, InTech, Available from: http://www.intechopen.com/books/principal-component-analysis/computing-and-updating-principal-components-of-discrete-and-continuous-point-sets

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the Creative Commons Attribution 3.0 License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen