LINEAR MANIFOLD APPROXIMATION BASED ON DIFFERENCES OF TANGENTS

Sofia Karygianni and Pascal Frossard

Signal Processing Laboratory (LTS4)
Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland
Email:{sofia.karygianni, pascal.frossard}@epfl.ch

ABSTRACT

In this paper, we consider the problem of manifold approximation with affine subspaces. Our objective is to discover a set of low dimensional affine subspaces that represents manifold data accurately while preserving the manifold's structure. For this purpose, we employ a greedy technique that partitions manifold samples into groups that can be well approximated by low dimensional subspaces. We start with considering each manifold sample as a different group and we use the difference of tangents to determine advantageous group mergings. We repeat this procedure until we reach the desired number of significant groups. At the end, the best low dimensional affine subspaces corresponding to the final groups constitute the manifold representation. Our experiments verify the effectiveness of the proposed scheme and show its superior performance compared to state-of-the-art methods for manifold approximation.

Index Terms— manifold, tangent space, affine subspaces, flats, greedy

1. INTRODUCTION

Signals often undergo transformations that appear to alter them critically. As a result, two transformed versions of a signal can appear significantly distinct, especially to they "eyes" of a computer. They are however representations of essentially the same entity. Invariance to transformations becomes crucial for effectively categorizing and recognizing signals correctly. Manifolds are often employed to achieve a signal description that is transformation invariant. According to the manifold model, the transformed versions of the same N-dimensional signal lie on a low dimensional structure embedded in $\Re^N$, whose dimensionality depends merely on the number of the transformation parameters. For example, an object can appear quite different depending on the image capture conditions. No matter how much different its images may seem though, they all belong to the manifold defined by the transformation parameters.

Eventhough manifolds seem appealing for transformation invariant applications, their unknown and usually strongly non-linear structure makes their manipulation quite tricky. One way to deal with this fact is to infer a global parametrization scheme, mapping the manifold data from the original space to a low-dimensional parametric space. The problem of unveiling such a parametrization is called manifold learning [1]. Usually, it is hard to discover a universal manifold representation that is always accurate as it demands that all the non-linearities of the manifold are well represented by only one mapping function. Therefore, instead of using just one global scheme, it is often preferable to employ a set of simpler structures to approximate manifold's geometry.

In our case we use a set of affine subspaces (flats) for approximating a manifold. Our objective is to compute a set of low dimensional flats that represents the data as accurately as possible and at the same time preserves the geometry of the underlying manifold. We relate the capability of a set of samples to be represented by a flat with the dimensionality of its affine hull, and we connect it to the samples' tangent spaces. Then, we use the difference of tangents to uncover groups of points that comply with the low dimensionality of flats. The partitioning is done in a greedy, bottom-up manner where each manifold sample is considered a different group at the beginning; groups are then iteratively merged until their number reduces to the desired value. The resulting scheme gives a promising performance compared to state-of-the-art manifold approximation techniques.

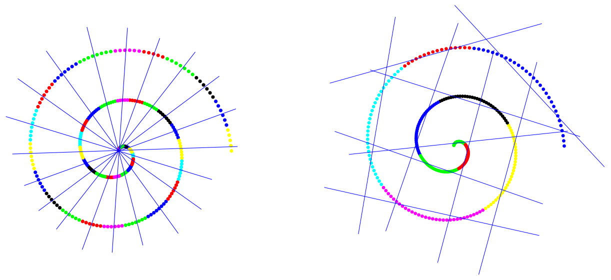
Manifold approximation with affine subspaces can be related to subspace clustering which is the general problem of representing data with flats. The proposed techniques usually use an iterative scheme alternating between data segmentation and subspace estimation [2], [3]. While being similar, the two problems are not identical, as in subspace clustering there is no guarantee that the manifold structure will be preserved by the final flats. Such an example is shown in Figure 1 where the Median k-flats algorithm [3] builds a set of flats that is not consistent with the underlying manifold geometry, as opposed to the outcome of our algorithm.

In manifold approximation the preservation of the manifold's structure is crucial. Works that manage to discover such flat-based manifold approximations are presented in [4] and in [5]. In [4], the authors introduce Hierarchical Divisive Clustering (HDC), a method for hierarchically partitioning the data, by dividing highly non-linear clusters. As a linearity measure they use the deviation between the euclidean and geodesic distances. In [5], the clustering is performed on a bottom-up manner, named Hierarchical Agglomerative Clustering (HAC), where again the geodesic distances are used to express the underlying manifold structure. Both methods are however shown to be inferior to our scheme, as the use of tangent spaces is proven to be more effective than the geodesic-distance based measures for manifold approximation.

The rest of the paper is organized as follows. In section 2, we give the problem formulation and in section 3, we present our algorithm in detail. In section 4, we describe the experimental setup and the results of our tests. Finally, in section 5, we discuss our conclusions.

2. PROBLEM FORMULATION

We consider the problem of approximating a d-dimensional manifold M, embedded into $\Re^N$, with a set of d-dimensional flats. The



(a) Median K-flats

(b) Our algorithm

Fig. 1. Example of a case where the minimization of the reconstruction error leads to a set of flats that does not follow manifold's geometry (a) as opposed to the outcome of our approach (b).

manifold is represented by the set of samples $D = \{x_k \in \Re^N, k \in [1, m]\}$ and the corresponding undirected and symmetric neighborhood graph $G(D, E)$. The objective is to partition D into l groups S_i, $i \in [1, l]$, each with corresponding affine hull A_i, representative d-dimensional flat P_i and subgraph $G_i = G(S_i, E_i)$ where $E_i = \{a_{wq} \in E : x_q, x_w \in S_i\}$, such that

$$S^* = \arg\min_{S} \sum_{S_i \in S} dif(A_i, P_i) \tag{1}$$

subject to:

$$\cup_{i=1}^{l} S_i = D \tag{2}$$

$$S_j \cap S_i = \emptyset,\ \forall\, i \neq j \tag{3}$$

$$G_i \text{ is connected}, \forall i \tag{4}$$

The function $dif(A_i, P_i)$ measures the quality of the d-dimensional approximation of the A_i's by the P_i's. The additional constraints refer to the form of the final groups S_i, demanding that each sample belongs to exactly one group and that each S_i is connected in terms of the neighborhood graph $G(D, E)$.

The affine hull of a set of points is the lowest dimensional linear manifold that includes all the points in the set and is formally defined as : $A_i = \{x | x = \sum_{j=1}^{|S_i|} \alpha_j * x_j,\ \alpha_j \in \Re,\ \sum_{j=1}^{|S_i|} \alpha_j = 1\}$. Ideally we would like to end up with d-dimensional A_i's. In such a case, each A_i would coincide with the tangent space T_j at each of the samples x_j in S_i, since the tangent space is naturally the best local d-dimensional affine approximation. Therefore, sets with such affine hulls contain samples with equal or similar tangents and are well represented by flats close to the "mean" tangent space over the samples of the group. As a result, we can measure the quality of a d-dimensional approximation of a group's affine hull ($dif(A_i, P_i)$) by computing the sum of differences between the individual tangent spaces of the samples in the group and the "mean" tangent.

In particular, the tangent space T_j at a manifold sample x_j is a linear subspace of $\Re^N$ located at the sample. Tangents live hence, in the Grassman manifold $G_{N,d}$ [6], the space of d-dimensional linear subspaces of $\Re^N$. Therefore, the distance of two tangents T_j and T_i with bases $B_{T_j} \in \Re^{N \times d}$ and $B_{T_i} \in \Re^{N \times d}$ can be expressed as:

$$D^2(T_i, T_j) = d - tr(B_{T_i}^T B_{T_j} B_{T_j}^T B_{T_i}), \quad T_i, T_j \in G_{N,d} \tag{5}$$

where $D^2(T_i, T_j)$ is the projection metric, a commonly employed distance measure in $G_{N,d}$ [7]. The "mean" tangent $P_i \in G_{N,d}$ of a group S_i, the Karcher mean [8] of tangents, is then the flat that minimizes the sum of square distances from the tangents $T_j, \forall x_j \in S_i$ i.e.,

$$P_i = \arg\min_{P \in G_{N,d}} \sum_{x_j \in S_i} D^2(P, T_j) \tag{6}$$

The sum of differences between the individual tangent spaces of the samples in a group and the mean tangent of the group, $dif(A_i, P_i)$, can then be expressed as:

$$dif(A_i, P_i) = \sum_{x_j \in S_i} D^2(T_j, P_i)$$

where P_i is given from equation (6) and $D^2(T_j, P_i)$ is as in (5). Finally, we can rewrite the approximation problem of equation (1) as :

$$S^* = \arg\min_{S} \sum_{S_i \in S} \sum_{x_j \in S_i} D^2(T_j, P_i) \tag{7}$$

3. GREEDY MERGING BASED ON THE DIFFERENCE OF TANGENTS

Our approximation algorithm is based on grouping the samples according to their tangent spaces to minimize the cost function in (7). The method is divided in two main steps. At the first step, the goal is to compute the tangent spaces for every sample in the dataset. At the second step, the objective is to combine the samples into groups and estimate their representative flats until reaching the desired number of groups. The block diagram of the method is shown in figure 2 .

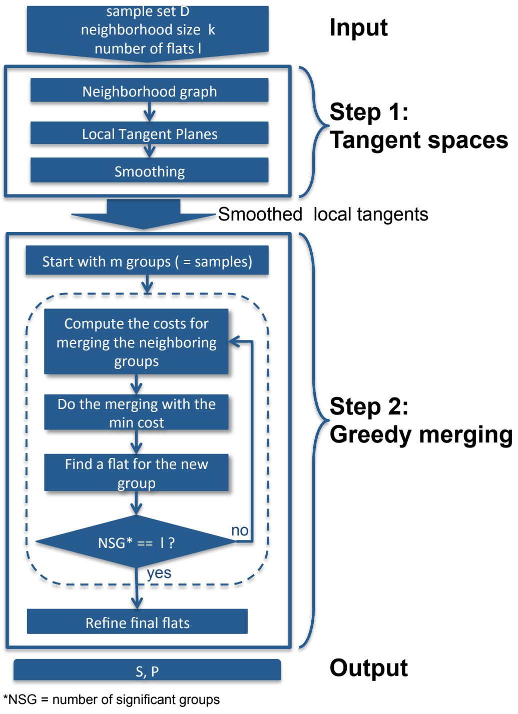


Fig. 2. The block diagram of the system

3.1. Tangent space

The first step of the algorithm consists of 3 distinct sub-steps:

1. Neighborhood graph construction
2. Tangent space computation
3. Smoothing

At first, we compute the undirected and symmetric neighborhood graph $G(D, E)$ by connecting every sample to its k nearest neighbors. The tangent space of each sample is then formed by the d eigenvectors that correspond to the d largest eigenvalues of the data matrix representing its neighborhood.

Finally, it might be the case that for some samples the resulting neighborhoods are not accurate enough, e.g., small k, noisy samples, outliers. We deal with this possible implication by adding a smoothing step to the process of tangent space computation. The smoothing is performed in the $G_{N,d}$ and is formulated as a weighted average tangent computation over each sample's neighborhood, similar to equation (6).

3.2. Greedy merging

Once the tangent spaces are computed, we proceed with solving the optimization problem presented in (7). In order to minimize the cost function we use a bottom-up technique: starting with m separate groups (each sample corresponds to a group represented by its own tangent space), we aim at reducing the total number of significant groups [1] to l by greedy merging. The optimal sequence of mergings could be inferred by a dynamic programming strategy [9] . While being straightforward, such an approach would be incompetent in terms of time complexity, especially in case of large sample sets. A greedy strategy [9] on the other hand, is an appealing alternative as its local nature in making decisions decreases significantly the computational time without large performance penalty.

In our case, we adopt a greedy scheme for deciding which groups to merge. The algorithm performs several iterations. At each iteration, there exists a set of possible mergings between the neighboring groups, so that the final S_i's fulfill inevitably condition (4), and the choice of the one that is performed depends on the additional cost that is introduced into (7). To be more specific, for a group S_i all mergings with its neighbors, defined as $NG_i = \{S_r : \exists x_w \in S_i,\ x_q \in S_r \text{ s.t } (x_w x_q) \in E\}$, are possible at each step. The cost introduced by a merging between S_i and $S_j \in NG_i$, represented with flats P_i and P_j, is the difference $dF(S_i, S_j)$ between the cost of the merged group S_{ij} and the sum of the costs for S_i and S_j before the merging:

$$\begin{aligned} dF(S_i,S_j) &= dif(A_{ij},P_{ij}) - dif(A_i,P_i) - dif(A_j,P_j) \\ &= \sum_{x_k\in S_{ij}} D^2(T_k,P_{ij}) - \sum_{x_k\in S_i} D^2(T_k,P_i) \\ &- \sum_{x_k\in S_j} D^2(T_k,P_j) \\ &= \sum_{x_k\in S_i} \left[D^2(T_k,P_{ij}) - D^2(T_k,P_i)\right] \\ &+ \sum_{x_k\in S_j} \left[D^2(T_k,P_{ij}) - D^2(T_k,P_j)\right] \end{aligned} \tag{8}$$

where P_{ij} is the flat representing the merged group S_{ij}, computed as the Karcher mean over the tangent spaces of the samples in S_{ij}, similar to equation (6):

$$P_{ij} = \underset{P\in G_{N,d}}{\arg\min} \sum_{x_k\in S_{ij}} D^2(P,T_k) \tag{9}$$

Since P_i and P_j are also both optimal for representing S_i and S_j in terms of minimizing the sum of square distances from the corresponding samples' tangents, $dF(S_i, S_j)$ will always be non-negative.

Unfortunately, it is too costly to compute all the new flats for all possible mergings. We compute instead an upper bound for $dF(S_i, S_j)$ that does not depend on P_{ij}. First, we substitute the difference of squares by its product equivalent and we use the reverse triangle inequality to bound $D(T_k, P_{ij}) - D(T_k, P_i)$ and $D(T_k, P_{ij}) - D(T_k, P_j)$ by $D(P_i, P_{ij})$ and $D(P_j, P_{ij})$, respectively. We have:

$$\begin{aligned} dF(S_i,S_j) \le D(P_i,P_{ij}) &\sum_{x_k\in S_i} [D(T_k,P_{ij}) + D(T_k,P_i)] \\ + D(P_j,P_{ij}) &\sum_{x_k\in S_j} [D(T_k,P_{ij}) + D(T_k,P_j)] \end{aligned}$$

By the triangle inequality, $D(T_k, P_{ij})$ can be upper bounded by $D(P_i, P_{ij}) + D(T_k, P_i)$ for $x_k \in S_i$ and by $D(P_j, P_{ij}) + D(T_k, P_j)$ for $x_k \in S_j$. After some simple mathematical manipulations and since $D(P_i, P_j)$ is greater or equal to both $D(P_i, P_{ij})$ and $D(P_j, P_{ij})$, we finally get that:

$$\begin{aligned} dF(S_i,S_j) &\le (|S_i| + |S_j|)D^2(P_i,P_j) \\ &+ 2D(P_i,P_j)\left[\sum_{x_k\in S_i} D(T_k,P_i) + \sum_{x_k\in S_j} D(T_k,P_j)\right] \end{aligned} \tag{10}$$

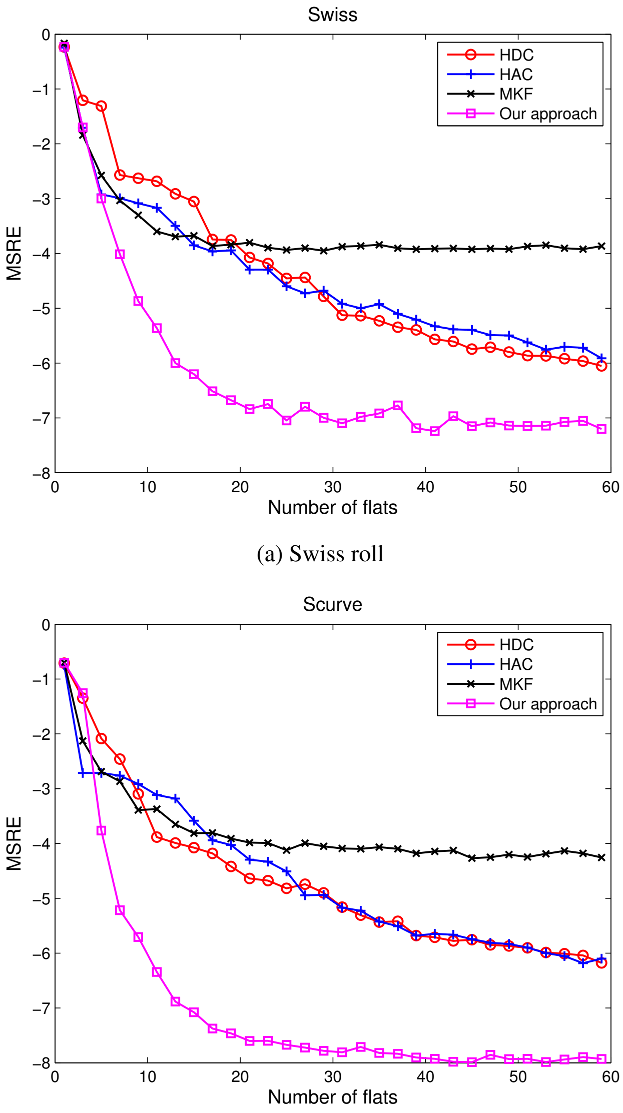
The costs for all possible mergings at each iteration are computed according to the formula in (10). The groups with the minimum estimated merging cost are then combined and the representative flat of the newly formed group is computed as in (9). The procedure is then repeated until we reach the desired number of significant flats.

At the end, we get the desired set of groups and their representative flats. However, these flats are the outcome of multiple optimization problems, and as such, they accumulate errors. In order to avoid this effect, we use as the final representative flats the subspaces spanned by the eigenvectors corresponding to the d largest eigenvalues of each group's data matrix.

4. EXPERIMENTAL RESULTS

We compare our scheme with HDC [4], HAC [5] and Median k-flats (MKF) [3] algorithms. For our experiments we use the Swiss roll and the S-curve dataset. The training set for both cases consists of 2000 points, randomly sampled from the manifolds. For testing, we use a new, randomly selected set of 5000 samples. The registration of the testing samples to the flats is done by majority voting over their k nearest neighbors in the training set. The neighborhood size k is set equal to 15 in the experiments. It is preferable to use low values for k, varying from 0.5% to 2% of the total number of samples, in order to avoid "short-circuit" effects that would distort manifold's structure. For the smoothing, we use gaussian weights, $a_r = (1/\sqrt{2\pi\sigma_j^2})exp(-\|x_r - x_j\|^2/(2\sigma_j^2))$ where σ_j is set equal to the 50% of the mean distance in the neighborhood of x_j. The optimization problem in (6) is solved using the Newton method as described in [6] for the Grassman manifold $G_{N,d}$.

[1] A group is significant when it contains more than 2% of the total number of samples.

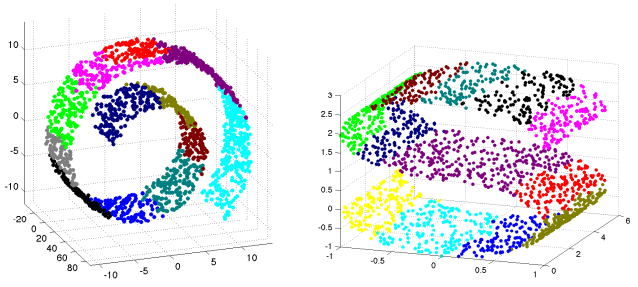


(a) Swiss roll

(b) S-curve

Fig. 3. Mean squared reconstruction error (MSRE) versus the number of flats. The error on the y-axis is shown in logarithmic scale.

The mean squared reconstruction error (MSRE) versus the number of flats is presented in Figure 3 where we can see that our scheme approximates better the manifold structure than the other approaches. The performance is higher even for small number of flats but the differences are more evident in the mid-range cases (number of flats from 15 to 30). After a certain number of flats the differences converge following the convergence of the individual MSREs. The effectiveness of our method is mainly accredited to the use of the difference of tangent spaces for measuring the linearity of sample sets instead of the geodesic-based criteria used previously. Moreover, an example of the final groups is shown in Figure 4 for the case of 12 flats, where we see that the structure of the manifold is correctly preserved by the proposed manifold approximation algorithm.



(a) Swiss roll

(b) S-curve

Fig. 4. The final groups with 12 flats.

5. CONCLUSIONS

We have presented a greedy, bottom-up method for approximating a manifold with low dimensional flats based on the difference of tangent spaces. The greedy optimization technique, in combination with the difference of tangents employed as a linearity measure, has been proven to be quite powerful in manifold approximation, outperforming existing manifold approximation approaches. The final low-dimensional representation of signals belonging to the manifold, can be used to achieve significant data compression. It can also be employed as a model for signal classification as the projections to the resulting flats can be considered a set of manifold based, class specific features.

6. REFERENCES

[1] Robert Pless and Richard Souvenir, “A survey of manifold learning for images,” *IPSJ Transactions on Computer Vision and Applications*, vol. 1, pp. 83–94, 2009.

[2] R. Cappelli, D. Maio, and D. Maltoni, “Multispace kl for pattern representation and classification,” *PAMI*, vol. 23, no. 9, pp. 977–996, September 2001.

[3] Teng Zhang, Arthur Szlam, and Gilad Lerman, “Median k-flats for hybrid linear modeling with many outliers,” *CoRR*, vol. abs/0909.3123, 2009.

[4] Ruiping Wang and Xilin Chen, “Manifold discriminant analysis,” *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 0, pp. 429–436, 2009.

[5] Wei Fan and Dit-Yan Yeung, “Locally linear models on face appearance manifolds with application to dual-subspace based classification,” in *CVPR ’06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, 2006, pp. 1384–1390, IEEE Computer Society.

[6] Alan Edelman, Tom s, A. Arias, Steven, and T. Smith, “The geometry of algorithms with orthogonality constraints,” *SIAM J. Matrix Anal. Appl*, vol. 20, pp. 303–353, 1998.

[7] J Hamm and D Lee, “Grassmann discriminant analysis: a unifying view on subspace-based learning,” *International conference on Machine learning*, Jan 2008.

[8] H. Karcher, “Riemannian center of mass and mollifier smoothing,” *Communications on Pure and Applied Mathematics*, vol. 30, no. 5, pp. 509–541, 1977.

[9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, 2001.