Data Fusion and Multi-Cue Data Matching by Diffusion Maps

Stéphane Lafon[1], Yosi Keller[2] and Ronald R. Coifman[2]

Abstract

Data fusion and multi-cue data matching are fundamental tasks of high-dimensional data analysis. In this paper, we apply the recently introduced diffusion framework to address these tasks. Our contribution is three-fold. First, we present the Laplace-Beltrami approach for computing density invariant embeddings which are essential for integrating different sources of data. Second, we describe a refinement of the Nyström extension algorithm called "geometric harmonics". We also explain how to use this tool for data assimilation. Finally, we introduce a multi-cue data matching scheme based on nonlinear spectral graphs alignment. The effectiveness of the presented schemes is validated by applying it to the problems of lip-reading and image sequence alignment.

Index Terms

Pattern matching, graph theory, graph algorithms, Markov processes, machine learning, data mining, image databases.

I. Introduction

The processing of massive high-dimensional data sets is a contemporary challenge. Suppose that a source s produces high-dimensional data $\{x_1, ..., x_n\}$ that we wish to analyze. For instance, each data point could be the frames of a movie produced by a digital camera, or the pixels of a hyperspectral image. When dealing with this type of data, the high-dimensionality is an obstacle for any efficient processing of the data. Indeed, many classical data processing algorithms have a computational complexity that grows exponentially with the dimension (this is the so-called "curse of dimensionality"). On the other hand, the source s may only enjoy a limited number of degrees of freedom. This means that most of the variables that describe each data points are highly correlated, at least locally, or equivalently, that the data set has a low intrinsic dimensionality. In this case, the high-dimensional representation of the data is an unfortunate (but often unavoidable) artifact of the choice of sensors or the acquisition device. Therefore it should be possible to obtain low-dimensional representations of the samples. Note that since the correlation between variables might only be local, classical global dimension reduction methods like Principal Component Analysis and Multidimensional Scaling do not provide, in general, an efficient dimension reduction.

First introduced in the context of manifold learning, eigenmaps techniques [1], [2], [3], [4] are becoming increasingly popular as they overcome this problem. Indeed, they allow one to perform a nonlinear reduction of the dimension by providing a parametrization of the data set that preserves neighborhoods. However, the new representation that one obtains is highly sensitive to the way the data points were originally sampled. More precisely, if the data are assumed to approximately lie on a manifold, then the eigenmap representation depends on the density of the points on this manifold [5]. This issue is of critical importance in applications as one often needs to *merge data* that were produced by the same source but acquired with different devices or sensors, at various sampling rates and possibly on different occasions. In that case, it is necessary to have a canonical representation of the data that retains the intrinsic constraints of the samples (e.g. manifold geometry) regardless of the particular distribution of the datasets sampled by different devices.

[1]Google Inc., stephane.lafon@gmail.com
[2]Department of Mathematics, Yale University, {yosi.keller, coifman-ronald}@yale.edu.

Another important issue is that of *data matching*. This question arises when one needs to establish a correspondence between two data sets resulting from the same fundamental source. For instance, consider the problem of matching pixels of a stereo image pair. One can form a graph for each image, where pixels constitute the nodes, and where edges are weighted according to the local features in the image. The problem now boils down to matching nodes between two graphs. Note that this situation is an instance of multi-sensor integration problem, in which one needs to find the correspondence between data captured by different sensors. In some applications, like fraud detection, synchronizing data sets is used for detecting discrepancies rather than similarities between data sets.

The out-of-sample extension problem is another aspect of the data fusion problem. The idea is to extend a function known on a training set to a new point using both the target function and the geometry of the training domain. The new point and the corresponding value of the function can then be assimilated to the training set. This is an essential component in any scheme that agglomerates knowledge over an initial data set and then applies the inferred structure to new data. Recently, Belkin *et al* have developed a solution to this problem via the concept of manifold regularization [6]. Earlier, several authors used the Nyström extension procedure in the Machine Learning context [7], [8] in order to extend eigenmap coordinates. In both cases, the question of the scale of the extension kernel remains unanswered. In other words, given an empirical function on a data set, to what distance to the training set can this function be extended ? In particular, given the spectral embedding of the data set, which kernel should be used to extend it?

By relating the frequency content of the target function on the training set to the extrinsic Fourier analysis, Coifman *et al* provide an answer to this question [9]. They developed the idea of "geometric harmonics" based on the Nyström extension at different scales, providing a multiscale extension scheme for empirical functions. We apply this concept to the extension of spectral embeddings and show that the extension has to be conducted using a specially designed kernel which differs from the eigenmap kernel.

In this article, we show that the questions discussed above can be efficiently addressed by the general diffusion framework introduced in [5], [10], [11]. The main idea is that, just like for eigenmaps methods, eigenvectors of Markov matrices can be used to embed any graph into a Euclidean space and achieve dimension reduction. Building on these ideas, the contribution of this paper is three-fold:

- First, we show that by carefully normalizing the Markov matrix, the embedding can be made invariant to the density of the sampled data points, thus solving the problem of data fusion encountered with other eigenmaps methods.
- Then, we address the problem of out-of-sample extension, and we explain how to adaptively extend empirical functions to new samples using the geometric harmonics. In particular this allows us to extend the diffusion coordinates to new data points.
- Last, we take advantage of the density-invariant representation of data sets provided by the diffusion coordinates to derive a simple data matching algorithm based on geometrical embeddings alignment.

The proposed scheme is experimentally verified by applying it to visual data analysis. First, we address the problem of automatic lip-reading by embedding the lips images using the Laplace-Beltrami eigenfunctions and deriving an automatic lip-reading scheme where new data is assimilated using geometric harmonics. Second, we demonstrate the multi-cue data matching aspect of our work by matching image sequences corresponding to similar head motions.

This paper is organized as follows: we start by recalling the diffusion framework, and the notion of diffusion maps in Section II-A. We then explain in Section II-B how to normalize the diffusion kernel in order to separate the geometry (constraints) of the data from the distribution of the points. We describe the out-of-sample extension procedure via the geometric harmonics in Section II-C and present a nonlinear algorithms for matching two data sets in Section II-D. Last, we illustrate these ideas by applying it to lip-reading and sequence alignment in Section III.

II. The diffusion framework

We start by reviewing the density-invariant embedding and out-of-sample extension schemes (previously introduced in [5] and [9]) in Sections II-B and II-C, respectively. To exemplify their applicability to high-dimensional data processing and learning, we apply them to derive a novel high-dimensional data alignment algorithm in Section II-D.

A. Diffusion maps and diffusion distances

Let $\Omega = \{x_1, ..., x_n\}$ be a set of n data points. In this section, we recall the diffusion framework as described in [5], [12], [13]. The main point of this set of techniques is to introduce a useful metric on data sets based on the connectivity of points within the graph of the data, and also to provide coordinates on the data set that reorganize the points according to this metric.

The first step in our construction is to view the data points $\Omega = \{x_1, ..., x_n\}$ as being the nodes of a symmetric graph in which any two nodes x_i and x_j are connected by an edge. The strength of this connection is measured by a non-negative weight $w(x_i, x_j)$ that reflects the similarity between x_i and x_j. The very notion of similarity between two data points is completely application-driven. In many situations however, each data point is a collection of continuous numerical measurements and, maybe after rescaling some of the features, it can be thought of as a point in a Euclidean feature space. In this case, similarity can be measured in terms of closeness in this space, and it is custom to weight the edge between x_i and x_j by $\exp(-\|x_i - x_j\|^2/\varepsilon)$, where $\varepsilon > 0$ is a scale parameter. This choice corresponds to the belief that the only relevant information lies in local distance measurements. Indeed, x_i and x_j will be numerically connected if they are sufficiently close. In diffusion kernels, graphs represent the structures of the input spaces, and the vertices are the objects to be classified. In addition, Belkin and Niyogi [2] explain that, in the case of a data set approximately lying on a submanifold, this choice corresponds to an approximation of the heat kernel on the submanifold. Last, in [5], it is shown that any weight of the form $h(\|x_i - x_j\|^2)$ (where h decays sufficiently fast at infinity) allows to approximate the heat kernel.

More generally, we allow ourselves to consider arbitrary weight functions $w(\cdot, \cdot)$ that verify the following two conditions[1], for all x and y in Ω:

- it is symmetric: $w(x, y) = w(x, y)$,
- it is pointwise non-negative: $w(x, y) \geq 0$.

This level of generality allows to take into account the case when data points are represented by a collection of categorical features. In this situation, it can be useful to employ a Gaussian kernel with a Hamming distance. But rather than to give a list of recipes, we would like to underline the fact that the choice of the weight function *should be entirely application-driven*. The weight function or kernel describes the first-order interaction between the data points as it defines the nearest neighbor structures in the graph. It should capture a notion of similarity as meaningful as possible with respect to the application, and therefore could very well take into account any type of prior knowledge on the data. The analysis of the data provided by the diffusion techniques depends heavily on the choice of the weight function. Last, note that the only real requirement for our technique to be applicable is to be able to define a *local* notion of similarity between the point. In other words, one must be able to answer the question of whether two points are (very) similar or not. This is a much simpler question than having to define a *global* distance between all pairs of points.

Following a classical construction in spectral graph theory [15], namely the normalized graph Laplacian, we now create a random walk on the data set Ω by forming the following kernel:

$$p_1(x, y) = \frac{w(x, y)}{d(x)},$$

where $d(x) = \sum_{z \in \Omega} w(x, z)$ is the degree of node x.

[1]Since $w(\cdot, \cdot)$ is supposed to represent the similarity between data points, it will be fair to assume that $w(x, x) > 0$

Since we have that $p_1(x,y) \geq 0$ and $\sum_{y\in\Omega} p_1(x,y) = 1$, the quantity $p_1(x,y)$ can be interpreted as the probability for a random walker to jump from x to y in a single time step. If P is the $n \times n$ matrix of transition of this Markov chain, then taking powers of this matrix amounts to running the chain forward in time. Let $p_t(\cdot,\cdot)$ be the kernel corresponding to the t^{th} power of the matrix P. In other words, $p_t(\cdot,\cdot)$ describes the probabilities of transition in t time steps.

The asymptotic behavior of this random walk has been used to find clusters in the data set [15], [16], [17], where the first non-constant eigenfunction is used as a classification function into two clusters. This was justified as a relaxation of a discrete problem of finding an optimal cut in a graph [16]. This approach was later generalized to using more eigenvectors in order to compute a larger number of clusters (see for instance [18], [19], [13]). Several papers form machine learning (in particular [14]) have underlined the connections and applications of the graph Laplacian to machine learning. Within the manifold learning community, the first few eigenvectors of this Markov chain have been employed for dimensionality reduction. In [20], [2] Belkin and Niyogi showed that when data is uniformly sampled from a low-dimensional manifold, the first few eigenvectors of P are discrete approximations of the eigenfunctions of the Laplace-Beltrami operator on the manifold, thus providing a mathematical justification for their use in this case.

If the graph is connected, then for $t = +\infty$ this Markov chain is governed by a unique stationary distribution ϕ_0 (see appendix I), which means that for all x and y,

$$\lim_{t\to+\infty} p_t(x,y) = \phi_0(y)\,.$$

The vector ϕ_0 is the top left eigenvector of P, *i.e.*, $\phi_0^T P = \phi_0^T$, and it can be verified that $\phi_0(y)$ is given by

$$\phi_0(y) = \frac{d(y)}{\sum_{z\in\Omega} d(z)}\,.$$

The pre-asymptotic regime is governed according to the following eigendecomposition [12]:

$$p_t(x,y) = \sum_{l\geq 0} \lambda_l^t \psi_l(x)\phi_l(y)\,, \tag{1}$$

where $\{\lambda_l\}$ is the sequence of eigenvalues of P (with $|\lambda_0| \geq |\lambda_1| \geq ...$) and $\{\phi_l\}$ and $\{\psi_l\}$ are the corresponding biorthogonal left and right eigenvectors (see appendix II for a proof). Furthermore, because of the spectrum decay, only a few terms are needed to achieve a given relative accuracy $\delta > 0$ in the previous sum.

Unifying ideas from Markov chains and potential theory, the *diffusion distance* between two points x and z was introduced in [12], [5] as

$$D_t^2(x,z) = \sum_{y\in\Omega} \frac{(p_t(x,y) - p_t(z,y))^2}{\phi_0(y)}\,. \tag{2}$$

This quantity is simply a weighted L^2 distance between the conditional probabilities $p_t(x,\cdot)$, and $p_t(z,\cdot)$. These probabilities can be thought of as features attached to the points x and z, and they measure the influence or interaction of these two nodes with the rest of the graph.

By increasing t, one propagates the local or short-term influence of each node to its nearest neighbors, and this means that t also plays the role of a scale parameter. The comparison of these conditional probabilities introduces a notion of proximity that accounts for the connectivity of the points in the graph. In particular, unlike the shortest path, or geodesic distance, this metric is robust to noise as it involves an integration along all paths of length t starting from x or z. Empirical evidence supporting this claim is provided in [13]. The diffusion distance incorporates the notions of mixing time and clusterness used in classical graph theory [21].

The connection between the diffusion distance and the eigenvectors goes as follows (see appendix II):

$$D_t^2(x,z) = \sum_{l\geq 1} \lambda_l^{2t}(\psi_l(x) - \psi_l(z))^2 \,. \tag{3}$$

Note that ψ_0 does not appear in the sum because it is constant. This identity means that the right eigenvectors can be used to compute the diffusion distance. The diffusion distance therefore generalizes the use of the eigenvectors for finding bottlenecks and clusters in the graph [21], and extends this approach by taking into account more than just the second largest eigenvalue.

Furthermore, and as mentioned before, because of the spectrum decay, only a few terms are needed to achieve a given relative accuracy $\delta > 0$ in the previous sum. Let $m(t)$ be the number of terms retained, and define the diffusion map

$$\Psi_t : x \longmapsto \left(\lambda_1^t \psi_1(x), \lambda_2^t \psi_2(x), \ldots, \lambda_{m(t)}^t \psi_{m(t)}(x)\right)^T \,. \tag{4}$$

This mapping provides coordinates on the data set Ω, and embeds the n data points into the Euclidean space $\mathbb{R}^{m(t)}$. In addition, the spectrum decay is the reason why dimension reduction can be achieved. This method constitutes a universal and data-driven way to represent a graph or any generic data set as a cloud of points in a Euclidean space. We also obtain a complete parametrization of the data that captures relevant modes of variability. Moreover, the dimension $m(t)$ of the new representation only depends on the properties of the random walk on the data, and not on the number of features of the original representation of the data. In particular, if we increase t, then $m(t)$ decreases and we capture larger-scale structures in the data.

B. Data merging using the Laplace-Beltrami normalization

We now direct our attention to the case when the original data points $\Omega = \{x_1, ..., x_n\}$ are assumed[2] to approximately lie on a submanifold $\mathcal{M}$ of $\mathbb{R}^d$. The so called "manifold model" holds for a large variety of situations, such as when the data is produced by a source controlled by a few free continuous parameters. For instance, consider the rotation of a human head and the lips motion of a speaker. We will study these examples later in this paper.

On the manifold $\mathcal{M}$, the data points were sampled with a density $q(\cdot)$ that may reflect some important aspect of the phenomenon that generated the data. For instance, as described in [12], for some data sets, the density is related to the free energy surface that governs the samples. On the other hand, the density may depend on the acquisition process and may be unrelated to intrinsic geometry or dynamics of the underlying phenomenon. In this situation, the distribution of the points is an artifact of the sampling process, and consequently, any "good" representation of the data should be invariant to the density.

Classical eigenmap methods provide an embedding that combines the information of both the density and geometry. For instance, with the Laplacian eigenmaps [2], one starts by forming the graph with Gaussian weights $w_\varepsilon(x,y) = \exp(-\|x-y\|^2/\varepsilon)$, and then constructs the random walk as described in the previous section. The eigenvectors are then used to embed the data set into a Euclidean space. It was shown in [5] that in the large sample limit $n \to +\infty$ and small scale $\varepsilon \to 0$, the eigenvectors tend to those of the Schrödinger operator $\Delta + E$, where Δ is the Laplace-Beltrami operator on $\mathcal{M}$, and E is a scalar potential that depends on the density q. As a consequence, the Laplacian eigenmaps representation of the data heavily depends on the density of the data points. In particular, it makes it impossible to fuse two data sets obtained from the same sensors but with different densities.

In order to solve this problem, we suggest to renormalize the Gaussian edge weights $w_\varepsilon(\cdot,\cdot)$ with an estimate of the density and to form the random walk on this new graph. This is summarized in Algorithm 1.

[2]Note that the density normalization that we describe in this section can be applied to more general structures such as a cloud of points. In this case, the diffusion coordinates will be invariant to the density of the points within this cloud.

Algorithm 1 Approximation of the Laplace-Beltrami diffusion

1: Start with a rotation-invariant kernel $w_\varepsilon(x,y) = h\left(\frac{\|x-y\|^2}{\varepsilon}\right)$.

2: Let

$$q_\varepsilon(x) \triangleq \sum_{y\in\Omega} w_\varepsilon(x,y)\,,$$

and form the new kernel

$$\widetilde{w}_\varepsilon(x,y) = \frac{w_\varepsilon(x,y)}{q_\varepsilon(x)q_\varepsilon(y)}\,. \tag{5}$$

3: Apply the normalized graph Laplacian construction to this kernel, *i.e.*, set

$$d_\varepsilon(x) = \sum_{z\in\Omega} \widetilde{w}_\varepsilon(x,y)\,,$$

and define the anisotropic transition kernel

$$p_\varepsilon(x,y) = \frac{\widetilde{w}_\varepsilon(x,y)}{d_\varepsilon(x)}\,.$$

Let P_ε be the transition matrix with entries $p_\varepsilon(\cdot,\cdot)$. The asymptotics for P_ε are given in the following theorem.

Theorem 1: In the limit of large sample and small scales, we have

$$\lim_{\varepsilon\to 0}\lim_{n\to+\infty} \frac{I-P_\varepsilon}{\varepsilon} = \Delta\,.$$

In particular, the eigenvectors of P_ε tend to those of the Laplace-Beltrami operator on $\mathcal{M}$. We refer to [5] for a proof. A similar analysis for the case of a uniform density $q \equiv 1$ is provided in [2], [22].

This result shows that the diffusion embedding that one obtains from an appropriately renormalized Gaussian kernel does not depend on the density q of the data points of $\mathcal{M}$. This algorithm allows one to successfully capture the nonlinear constraints governing the data, independently from the distribution of the points. In other words, it separates the geometry of the manifold from the density.

C. Out-of-sample extension and the geometric harmonics

In most applications, it is essential to be able to extend the low-dimensional representation computed on a training set to new samples. Let Ω be a data set and Ψ_t be its diffusion embedding map. We now present the geometric harmonic scheme that allows us to extend Ψ_t to a new data set $\widetilde{\Omega}$. Since we need to relate the new samples to the training set, we will assume that Ω is a subset of a Euclidean space $\mathbb{R}^d$.

As mentioned in the introduction, the Nyström extension method is a popular technique employed in the machine learning community [7], [8] for the extension of empirical functions from the training set to new samples. As we discuss later, this method suffers from several drawbacks, and the scheme that we present in this section aims at solving these problems.

For the sake of completeness, we first recall the idea of Nyström extension [23]. We then point out its weaknesses, present our geometric harmonics extension scheme and explain how it solves the problems of the Nyström extension. Let $\sigma > 0$ be a scale of extension, and consider the eigenvectors and eigenvalues of a Gaussian kernel[3] of width σ on the training set Ω:

$$\mu_l\varphi_l(x) = \sum_{y\in\Omega} e^{-\|x-y\|^2/\sigma^2}\varphi_l(y) \text{ where } x\in\Omega\,.$$

[3]In order to simplify our presentation of the extension algorithm, we choose to work with a Gaussian kernel. In general, one can use any symmetric kernel with an exponential decay.

Since the kernel can be evaluated in the entire space, it is possible to take any $x \in \mathbb{R}^d$ in the right-hand side of this identity. This yields the following definition of the Nyström extension of φ_l from Ω to $\mathbb{R}^d$:

$$\overline{\varphi}_l(x) \triangleq \frac{1}{\mu_l} \sum_{y \in \Omega} e^{-\|x-y\|^2/\sigma^2} \varphi_l(y) \text{ where } x \in \mathbb{R}^d \,. \tag{6}$$

Note that φ_l is being extended to a distance proportional to σ from the training set Ω. Beyond this distance, the extension numerically vanishes.

We now know how to extend the eigenfunctions of the kernel, and since these eigenfunctions form a basis of the set of functions on the training set, any function f on the training set can be decomposed as the sum

$$f(x) = \sum_l \langle \varphi_l, f \rangle \, \varphi_l(x) \text{ where } x \in \Omega \,,$$

and we can define the Nytström extension of f to the rest of R^d to be

$$\overline{f}(x) \triangleq \sum_l \langle \varphi_l, f \rangle \, \overline{\varphi_l}(x) \text{ where } x \in \mathbb{R}^d \,. \tag{7}$$

This scheme seems very attractive, but it raises the question of the choice of the kernel of extension. In our exposition above, we considered a Gaussian of width σ, which implies that functions will be extended to a distance proportional to σ (the extension numerically vanishes beyond a multiple of this distance). Classically (see [7], [8]), when extending eigenmaps, the kernel being used for the extension is the same as the one employed for the computation of the eigenmaps on the training set. The focal point of the extension scheme that we now present is precisely to contradict this approach. Indeed, when computing the diffusion embedding or any other type of Laplacian eigenmap, one strives for using as small a scale $\sqrt{\varepsilon}$ as possible. The reason behind this is that, as shown in Theorem 1 and in [2], [22], [5], in the limit of small scales, the diffusion maps approximate the eigenvectors of the Laplace-Beltrami, allowing to capture the geometry of the underlying structure of the data set (such as the manifold geometry if there is an underlying manifold). On the contrary, when extending the diffusion coordinates off the training set, it is our interest to extend them as far as possible in order to maximize their generalization power. This has two consequences:

- The scale σ of the kernel used for extending should be as large as possible.
- This scale should not be the same for all functions that we are trying to extend. Indeed, we expect the scale of extension to be related to the complexity of the function to be extended. Low-complexity functions should be easy to extend very far from the training set. For instance the constant function on Ω is the simplest function on the training set, and should be extendable to the entire space R^d. On the contrary, a function with wild variations on Ω should have a limited range of extension, as their values off the training set are more difficult to predict.

These two observations give rise to the idea of adapting the scale of extension (and hence the kernel) to the function f to be extended. Therefore, all we need now is a criterion for determining the maximum scale of extension for f. To this end, fix $\sigma > 0$, and observe that in Equation 6, $\mu_l \to 0$ as $l \to +\infty$, which implies that the Nyström extension scheme described by Equation 7 is ill-conditioned. Of course, we can circumvent this problem if, in the same sum, we only retain the terms corresponding to μ_0/μ_l smaller than a given threshold $\eta > 0$:

$$\overline{f}(x) \triangleq \sum_{l : \mu_0 < \eta \mu_l} \langle \varphi_l, f \rangle \, \overline{\varphi_l}(x) \text{ where } x \in \mathbb{R}^d \,. \tag{8}$$

This way, the extension procedure has a condition number less than to η, and this variable plays the role of a regularization parameter. However, $\overline{f}$ and f no longer coincide on Ω, which means that $\overline{f}$ is no longer an extension of f. This is precisely the basis of decision about the scale σ: if it turns out that the difference between $\overline{f}$ and f on Ω is still acceptable (as measured by the reconstruction error), then this means that

f is extendable at a distance σ from Ω. Otherwise, it means that σ needs to be reduced. Indeed, if we decrease the value of σ, then the kernel of extension becomes finer, and its eigenvalues will decay more slowly. This allows the sum in Equation 8 to contain more terms, and $\overline{f}$ to be a better approximation of f on Ω. This geometric harmonics technique formalizes these observations into a scheme presented in Algorithm 2.

Algorithm 2 Multiscale extension scheme of diffusion coordinates via geometric harmonics

1: Let $\Omega \subset \mathbb{R}^d$ be the training set and $f = \psi_i : \Omega \to \mathbb{R}$ be the diffusion coordinate to be extended ($1 \leq i \leq m(t)$). Choose a condition number $\eta > 0$ and an admissible error $\tau > 0$.

2: Choose an initial (large) scale of extension $\sigma = \sigma_0$.

3: Compute the eigenfunctions of the Gaussian kernel with width σ on the training set Ω:

$$\mu_l \varphi_l(x) = \sum_{y \in \Omega} e^{-\|x-y\|^2/\sigma^2} \varphi_l(y) \text{ where } x \in \Omega\,,$$

and expand f on this orthonormal basis (on the training set Ω):

$$f(x) = \sum_{l \geq 0} c_l \varphi_l(x) \text{ where } x \in \Omega\,.$$

4: Compute the error of reconstruction on the training set that one obtains by retaining only the coefficients such that $\eta > \mu_0/\mu_l$ in the sum above:

$$Err = \left(\sum_{l:\, \eta \leq \mu_0/\mu_l} |c_l|^2 \right)^{\frac{1}{2}}\,.$$

If $Err > \tau$ then divide σ by 2 and go back to point 3. Otherwise continue.

5: For each l such that $\eta > \mu_0/\mu_l$, extend φ_l via the Nyström procedure:

$$\overline{\varphi}_l(x) \triangleq \frac{1}{\mu_l} \sum_{y \in \Omega} e^{-\|x-y\|^2/\sigma^2} \varphi_l(y) \text{ where } x \in \mathbb{R}^d\,,$$

and define the extension $\overline{f}$ of f to be

$$\overline{f}(x) \triangleq \sum_{l \geq 0} c_l \overline{\varphi}_l(x) \text{ where } x \in \mathbb{R}^d\,.$$

To summarize our ideas, if we increase the scale of extension, then the error of reconstruction on Ω will increase. Hence, the reconstruction error limits the maximal extension range. In fact, this limitation can be regarded as relating the complexity of the function on the training set to the distance to which it can be extended off this set. Here, the notion of complexity is measured in terms of frequency content on the training domain. For instance, a constant function has almost no complexity and one should be able to extend it in the entire space. If the number of oscillations of this function increases, then the distance to which one can extend it gets smaller. This illustrated on Figure 1. The geometric harmonics are therefore perfectly appropriate for extending the diffusion coordinates to new samples as higher-order and lower-order diffusion coordinates do not have the same number of oscillations.

D. Multi-cue alignment and data matching

The purpose of this section is to explain how the diffusion embedding can be efficiently used for data matching. Suppose that one has two data sets $\Omega_1 = \{x_1, ..., x_n\}$ and $\Omega_2 = \{y_1, ..., y_{n'}\}$ for which one would like to find a correspondence, or detect similar patterns and trends, or on the contrary, underline

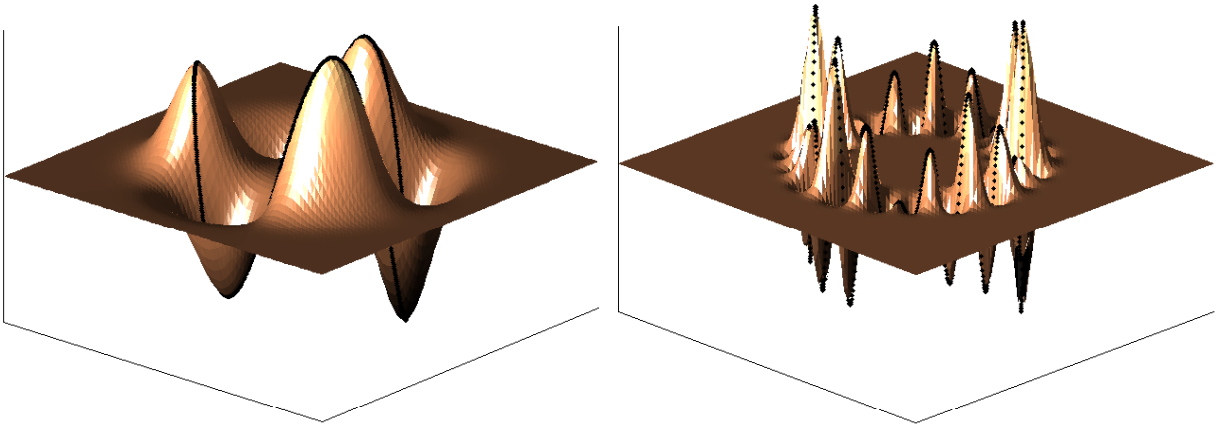


Fig. 1. Extension of two functions from the unit circle to $\mathbb{R}^2$. The function on the left is very smooth on the training set, and therefore can be extended far away from it. On the contrary, the function on the right oscillates much on the training set, and this limits its scale of extension.

their dissimilarity and detect anomalies. This type of task is very common in applications related to marketing, automatic machine translation, fraud detection or even counter-terrorism. However, working with the data in its original form can be quite difficult as the two sets typically consist of measurements of very different nature. For instance Ω_1 could be a collection of measurements related to wether in a given region, whereas Ω_2 could describe agriculture production in the same region. As a consequence, it is almost always impossible to directly compare the two data sets, simply because they might not be represented using the same type of features. The main idea that we introduce here is that the diffusion maps provide a canonical representation of data sets reflecting their intrinsic geometry. This new representation is based on the graph structure of a set, that is, the neighbor relationship between points, and not on their original feature representation. As a consequence, *instead of comparing the data sets in their original forms, it can be much more efficient to compare their embeddings*. In particular, if Ω_1 and Ω_2 are expected to have similar intrinsic geometry structures, then they should have similar embeddings.

There has been a body of work related to graph based manifold alignment. Gori et. al [24] align weighted and unweighted graphs by computing a 'signature' for each node that is based on repeated use of the invariant measure of different Markov chains defined on the data. The nodes/samples are then matched in two ways. First, in a one-by-one basis, where nodes with similar signatures are coupled. Second, in a globally optimal approach using a bipartite graph matching scheme. Ham et. al [25] align the manifolds, given a set of a-priori corresponding nodes or landmarks. A constrained formulation of the graph Laplacian based embeddings is derived by including the given alignment information. First, they add a term fixing the embedding coordinates of certain samples to predefined values. Both sets are then embedded separately, where certain samples in each set are mapped to the same embedding coordinates. Second, they describe a dual embedding scheme, where the constrained embeddings of both sets are computed simultaneously, and the embeddings of certain points in both datasets are constrained to be identical. The work of Bai et. al [26] presents a similar framework to our scheme. The ISOMAP algorithm is used to embed the nodes of the graphs corresponding to the aligned datasets, in a low-dimensional Euclidean space. The nodes are thus transformed into points in a metric space, and the graph-matching is recast as the alignment of point sets. A variant of the Scott and Longuet-Higgins algorithm is then used to find point correspondences. An approach to Many-to-Many alignment was presented in [27] by Keselman et. al. They aim to match corresponding clusters of nodes in both datasets, rather then match individual nodes. The datasets are embedded in a metric space using the Matousek embedding and sets of nodes are then aligned using the

Earth Mover's Distance, which is a distribution-based similarity measure for sets.

In the data alignment segment of our work, we resolve the alignment of datasets with a common low-dimensional manifold, but different densities, by incorporating the use of the density-invariant embedding. This issue was overlooked in previous works based on spectral embeddings [24], [25], [26], [27], although spectral and ISOMAPS embeddings are highly sensitive to the way the data points were originally sampled. Hence, the underlying assumption in [24], [25], [26], [27] that the low-dimensional embedding of datasets sharing a common low-dimensional manifold will be similar, might prove invalid.

In addition to dealing with the density issue, we present a semi-supervised algorithm for finding a one-to-one correspondence between two data sets. The scheme we introduce consists in aligning two graphs in a nonlinear fashion, based on a finite number of landmarks (matching points or nodes). The main idea is to lift each graph into the same diffusion space, and to align the resulting clouds of points using a simple affine matching[4]. The diffusion maps provide a nonlinear reduction of dimensionality, and therefore our scheme is appropriate for the alignment of high-dimensional data sets with low-intrinsic dimensionality. In addition, as explained in the previous sections, if we use the density-invariant diffusion maps, the alignment scheme will be insensitive to the different distributions of points of the two data sets.

As for the notations, suppose that we have $k < n, n'$ landmarks in each set, that is a sequence of k pairs $(x_{\sigma(1)}, y_{\tau(1)}), ..., (x_{\sigma(k)}, y_{\tau(k)})$ for which there is a known correspondence. This set of examples is the only prior information that we use in the algorithm. We assume that $x_{\sigma(1)} \neq x_{\sigma(2)} \neq ... \neq x_{\sigma(k)}$. The scheme given in Algorithm 3 computes a surjective function $g : \Omega_1 \to \Omega_2$ such that $g(x_{\sigma(1)}) = y_{\tau(1)}, ..., g(x_{\sigma(k)}) = y_{\tau(k)}$.

Algorithm 3 Nonlinear graph alignment

1: Start with k landmarks $(x_{\sigma(1)}, y_{\tau(1)}), ..., (x_{\sigma(k)}, y_{\tau(k)})$.

2: Compute the diffusion embeddings $\{\widetilde{x}_1, ..., \widetilde{x}_n\}$ and $\{\widetilde{y}_1, ..., \widetilde{y}_{n'}\}$ of Ω_1 and Ω_2 where, for each set, the time parameter was chosen so that $k-1$ eigenvectors are retained. In other words, $\widetilde{x}_i$ and $\widetilde{y}_j$ both live in $\mathbb{R}^{k-1}$.

3: Compute the affine function $f : \mathbb{R}^{k-1} \to \mathbb{R}^{k-1}$ that satisfies the landmark constraints:

$$f(\widetilde{x}_{\sigma(1)}) = \widetilde{y}_{\tau(1)}, ..., f(\widetilde{x}_{\sigma(k)}) = \widetilde{y}_{\tau(k)} \, .$$

4: Define the correspondence between Ω_1 and Ω_2 by

$$g(x_i) = \arg\min_{y \in \Omega_2} \{\|f(x_i) - y\|\} \, ,$$

where $x_i \in \Omega_1$,

The idea behind the scheme presented is to embed both data sets into the (same) diffusion space, and to use an affine alignment function f in the diffusion space. We assume that the choice of the kernels for computing the embeddings was already made by the user, and that they were selected in order to obtain meaningful graphs with respect to the application that the user has in mind. The number of eigenvectors used for the embedding is directly related to the number of landmarks, which in turns, represents the quantity of prior information for aligning. The larger the number of known constraints on the alignment, the larger the dimensionality of the aligning mapping. This is consistent with the fact that higher order eigenvectors capture finer structures. These observations pave the way for a general sampling theory for data sets. Indeed, the landmarks can be regarded as forming a subsampling of the original data sets. This subset determines the largest (or Nyquist) frequency used to represent the original set. This frequency is measured as the number of eigenvectors employed.

[4]We note that the alignment procedure can be automated for low-dimensional embeddings (up to R^3) by utilizing point matching schemes such as ICP [28] and Geometrical Hashing [29].

Note also that the affine function that we use for aligning induces a nonlinear mapping defined on lower dimensional embedding of the sets, and is even more nonlinear in the original space. It is possible to introduce more robustness to our scheme by embedding in a lower dimension than the number of landmarks, and to look for the best affine function that aligns the landmarks, where "best" is measured in a least-square sense.

III. Experimental results

A. Application to lip-reading

The validity of our approach is now demonstrated by applying it to lip-reading and sequence alignment, which are typical high-dimensional data analysis problems. From the statistical learning point of view, this example allows us to apply the ideas presented in the previous sections to three fundamental and related problems in the learning of high-dimensional data in general, and visual data in particular. First. we apply the diffusion framework to perform an efficient nonlinear dimensionality reduction. Second, we extend it to derive an intensity invariant embedding, essential for incorporating several data sources. Finally, we deal with the extension of a given embedding, computed on a given data set, to a new sample. This is the essence of a 'learning' schemes that associates knowledge obtained on a training set to a new set of samples.

Lip-reading has recently gained significant attention [30], [31], [32], [33], [34] and we now provide background and previous results in that field. The ultimate goal of lip-reading is to design human-like man-machine interfaces allowing automatic comprehension of speech, which in the absence of sound is denoted as lip-reading and the synthesis of realistic lip movement. The design of such a system involves three main challenges: first, the feature extraction, which aims at converting the images of the lips into a useful description, must be achieved with minimal preprocessing. Then, in order to be efficiently processed, the data must be transformed via a dimension reduction technique. Last, in order to assimilate new data for recognition, one must be able to perform data fusion.

Previous lip-reading schemes have mainly focused on the first two points. Concerning the feature extraction, some works [30], [34] analyze directly the intensity values of the input images, while others [35], [31] start by detecting curves and points of interest around the mouth whose locations are then used as features. The combination of audio-visual cues was used in [36] where the visual cues are the extracted lip contours which are tracked over time. We note that combining audio-visual is beyond the scope of this work and will be dealt by us in the future. Identifying, tracking and segmenting the lips is a difficult task and possible solutions include: active contours [37], probabilistic models [38] and the combination of multiple visual cues (shape, color and motion) [39] to name a few. In practice, one strives to use a simple preprocessing scheme as possible and in our scheme we employ a simple stabilization scheme discussed below.

Regarding the dimensionality reduction, several schemes have been used. Preliminary work employed linear algorithms such as the PCA and SVD subspace projections [35], [34]. For instance, Li *et al* [34] use a linear PCA scheme similar to the eigenfaces approach to face detection. Recognition is performed by correlating an input sequence with the eigenfeatures obtained from PCA. More recent schemes [30] utilize non-linear approaches such as the MDS [40]. Some of the techniques provide a general embedding framework for lipreading analysis [30], while others [34], [31] concentrate on a particular task such as phoneme or word identification. The work in [41] is of particular interest, since it is one of the first to explicitly formulate the lipreading problem as a "Manifold Learning" issue and tries to derive the inherent constraints embedded in the space of lip configurations. A Hidden Markov Model (HMM) is used to model a small number of words (names of four drinks) which define the Markov states and the manifold. The HMM is then used to recognize the drinks' names where the input is given by tracking the outer lips contour using Active Contours. Utilizing both audio and visual information significantly decreased the error rate, especially in noisy environments. Kimmel and Aharon [30] applied the MDS scheme to visual lips representation, analysis and synthesis. A set of lips images is aligned and embedded

in a two dimensional domain which is then sampled uniformly in the embedding domain to achieve uniform density. The pronunciation of each word is defined as a path over the embedding domain and used for visual speech recognition, by path matching. Lips motion synthesis is derived by computing the geodesic path over the embedding domain, where the start and end point are given as input. Anchor points in the low-dimensional embedding domain were then used to match the lips configurations of two different speakers.

Analysis of lip data constitutes an application where it is important to separate the set of nonlinear constraints on the data from the distribution of the points. As an illustration of the Laplace-Beltrami normalization as well as the out-of-sample extension scheme, we now describe an elementary experiment that paves the way to building automatic lip-reading machines, and more generally, machine learning systems.

We first recorded a movie of the lips of a subject reading a text in English. The subject was then asked to repeat each digit "zero", "one", ... , "nine" 40 times. A minimal preprocessing was applied to the recorded sequence. More precisely, it was first converted from colors to gray level (values between 0 and 1). Moreover, using a marker put at the tip of the nose of the speaker during the recording, we were able to automatically crop each frame into a rectangular area around the lips. Each of these new frames was then regarded as a point in $\mathbb{R}^{140\times 110}$, where 140×110 is the size of the cropped area.

The first data set, consisting of approximately 5000 frames, corresponds to the speaker reading the text. This set was used to learn the structures of the lip motion. More precisely, we formed a graph with Gaussian weights $\exp(-\|x_i - x_j\|^2/\varepsilon)$ on the edges between all pairs of points, where the distance $\|x_i - x_j\|$ was merely calculated as the Euclidean L^2 distance between frames i and j. The scale $\varepsilon > 0$ was chosen by looking at the distribution of the distances from each point to the other points. We selected $\sqrt{\varepsilon}$ such that each data point would be numerically connected with at least one other point in the graph. This value, which was found to be equal to 1000, turned out to make the graph of the data totally connected. The choice of this number was also coherent with the shape of the distribution of the distances (see Figure 2) in that, on average, each point is connected to a small fraction of the other points.

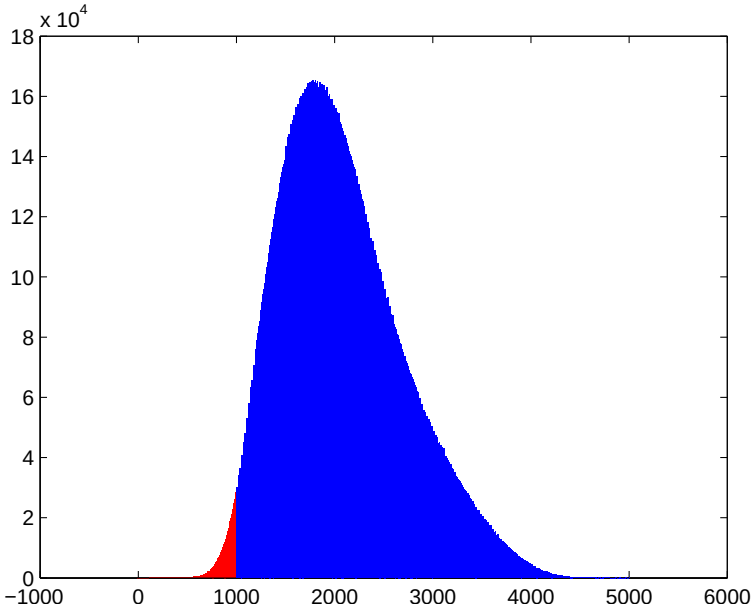


Fig. 2. The distribution of distances between all pairs of data points. The choice of the scale $\sqrt{\varepsilon} = 1000$ corresponds to having each data point connected to at least one other data point. The resulting graph happened to be totally connected. This histogram shows that the choice of this scale parameter leads to a sparse graph: each node is connected, on average, to a small number of other nodes.

We then renormalized the Gaussian weights using the Laplace-Beltrami normalization described in Section II-B. By doing so, our analysis focused on viewing the mouth as a constrained mechanical

system. In order to obtain a low-dimensional parametrization of these nonlinear constraints, we computed the diffusion coordinates on this new graph. The spectrum of the diffusion matrix is plotted on Figure 3 and the embedding in the first 3 eigenfunctions is shown on Figure 4.

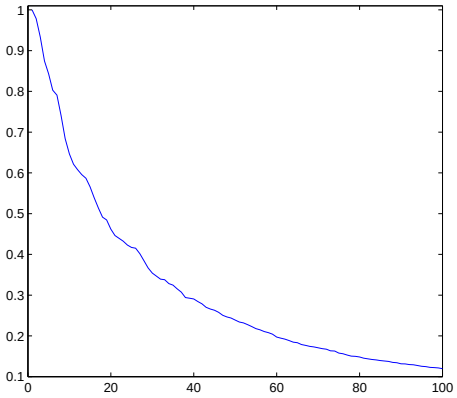


Fig. 3. The top 100 eigenvalues of the diffusion matrix for the lips data. The spectrum decays rapidly.

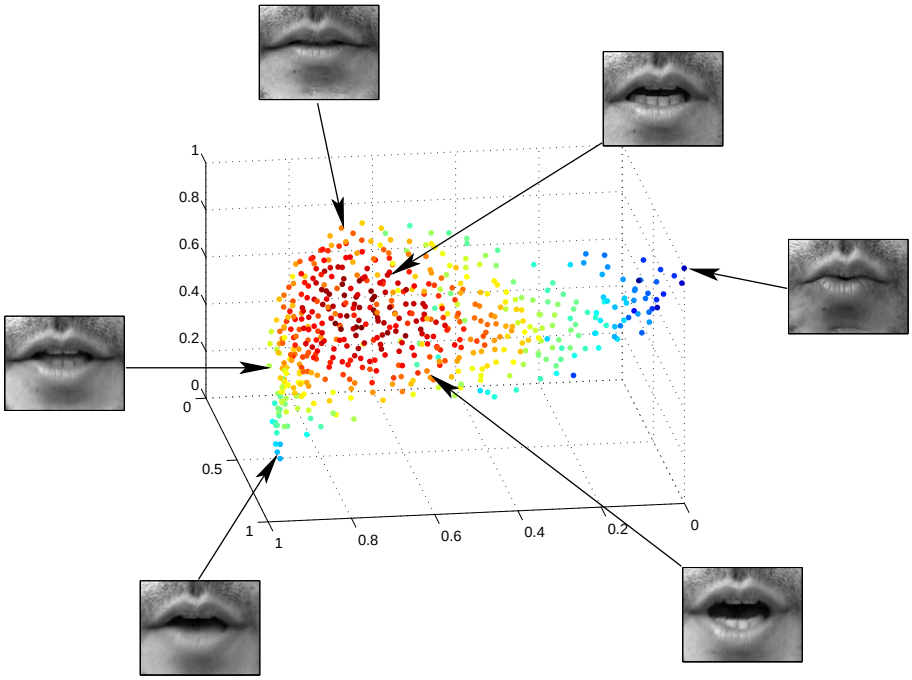


Fig. 4. The embedding of the lip data into the top 3 diffusion coordinates. These coordinates essentially capture two parameters: one controlling the opening of the mouth and the other measuring the portion of teeth that are visible.

The task we wanted to perform was isolated-word recognition on a small vocabulary. The example that we considered was that of identification of digits. Each word “zero”, “one”,..., “nine” is typically a sequence 25 to 40 frames that we need to project in the diffusion space[5]. In order to do so, we used the geometric harmonic extension scheme presented in Section II-C to extend each diffusion coordinate to the frames corresponding to the subject pronouncing the different digits. After this projection, each word can be viewed as a trajectory in the diffusion space. The word recognition problem now amounts to identifying trajectories in the diffusion space.

[5]Note that this second data set was *not* used to compute the diffusion maps.

We can now build a classifier based on comparing a new trajectory to a collection of labeled trajectories in a training set. We randomly selected 20 instances of each digit to form a training set, the remaining 20 being used as a testing set. In order to compare trajectories in the diffusion space, a metric is needed, and we chose to use the Hausdorff distance between two sets Γ_1 and Γ_2, defined as

$$d_H(\Gamma_1, \Gamma_2) = \max \left\{ \max_{x_2 \in \Gamma_2} \min_{x_1 \in \Gamma_1} \{\|x_1 - x_2\|\}, \max_{x_1 \in \Gamma_1} \min_{x_2 \in \Gamma_2} \{\|x_1 - x_2\|\} \right\} .$$

Although this distance does not use the temporal information, it has the advantage of not being sensitive to the choice of a parametrization or to the sampling density for either set Γ_1 and Γ_2. For a given trajectory Γ from the testing set, our classifier is a nearest-neighbor classifier for this metric, *i.e.*, the class of Γ is decided to be that of the nearest trajectory (for d_H) in the training set. The performance of this classifier averaged over 100 random trials is shown in Table I. In this case, the data set was embedded in 15 dimensions.

	"0"	**"1"**	**"2"**	**"3"**	**"4"**	**"5"**	**"6"**	**"7"**	**"8"**	**"9"**
zero	**0.93**	0	0	0.01	0	0	0.06	0	0	0
one	0	**1**	0	0	0	0	0	0	0	0
two	0.05	0	**0.88**	0.05	0.01	0	0.01	0	0	0
three	0.01	0	0.02	**0.93**	0	0	0.01	0.01	0.01	0.01
four	0	0	0.01	0.01	**0.97**	0	0	0.01	0	0
five	0	0	0	0.01	0	**0.84**	0.01	0.14	0	0.01
six	0.04	0	0	0.01	0	0	**0.92**	0.02	0	0.01
seven	0.02	0	0	0.04	0	0.07	0.10	**0.69**	0.05	0.03
eight	0	0.01	0	0	0	0.03	0.01	0.04	**0.77**	0.14
nine	0	0	0	0.02	0	0	0	0.02	0.12	**0.85**

TABLE I

CLASSIFIER PERFORMANCE OVER 100 RANDOM TRIALS. EACH ROW CORRESPONDS THE CLASSIFICATION DISTRIBUTION OF A GIVEN DIGIT OVER THEN 10 CLASSES. THE DATA SET WAS EMBEDDED IN 15 DIMENSIONS.

The classification error ranges from 0% to 31% with an average of 12.2%. The best classification rate is achieved for the word "one" which, in terms of visual information, stands far away from the other digits. In particular, typical sequences of "one" involve frames with a round open mouth, with no teeth visible (see first row of Figure 5). These frames essentially never appear for other digits. The worst classification job is for the word "seven" which seems to be highly confused with the words "five" and "six". As shown on Figure 5, typical instances of these words appear to be similar in that the central frames involve an open mouth with visible teeth. In the case of the "six" and "seven", teeth from the lower jaws are visible because of the "s" sound. Regarding the similarity between "five" and "seven", the "f" and "v" sounds translate into the lower lip touching the teeth of the upper jaw.

The accuracy that we obtain is comparable to former schemes [30], [41], while using significantly less preprocessing. For instance, in [30], the lips images are hand picked and stabilized using an affine motion model, while in [41] the contours of the lips are tracked by Active Contours. Our lips images are acquired by taping a continuous 5 minutes sequence and a simple cropping is performed to compensate for translations. We note that the above comparison is qualitative rather than quantitative, as the different schemes were applied to different datasets that are not publicly available.

B. Synchronization of head movement data

We now illustrate the concept of graph alignment as well as the algorithm presented in Section II-D. We recorded 3 movies of subjects wearing successively a yellow, red and black mask. Each subject was asked to move their head in front of the camcorder. We then considered the three sets consisting of all frames of each movie. Let YELLOW, RED and BLACK denote these sets. Our goal was to synchronize

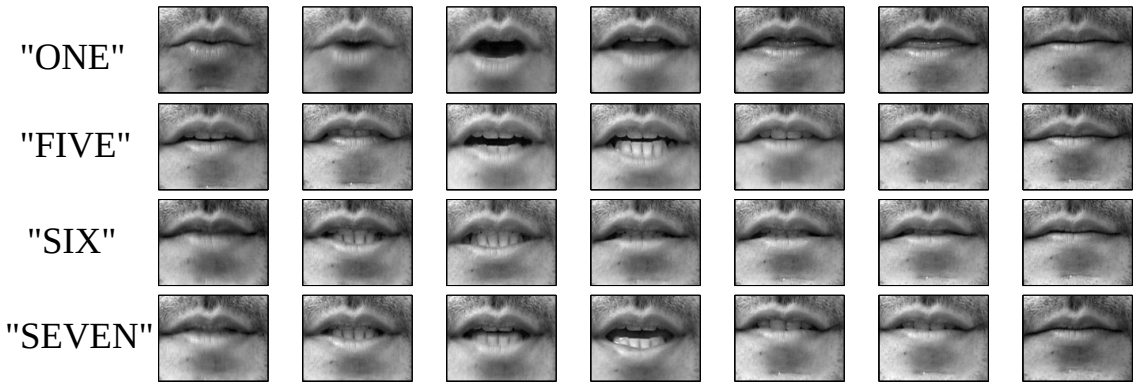


Fig. 5. Typical frames for the words “one”, “five”, “six”, “seven”.

the movements of the different masks by aligning the 3 diffusion embeddings. The objective of this experiment was twofold

- We first wanted to illustrate the importance of having a coordinate system capturing the intrinsic geometry of data sets. The intrinsic geometry is the basis of our alignment scheme: the key point is that, as we will show, all three sets exhibit approximately the same intrinsic geometry, and that the diffusion coordinates parameterize this geometry. It is to be noted that working directly in image space would be highly inefficient since any picture of the red or black mask is at a large distance from the set of pictures of the yellow mask (this is a straight consequence of the high dimensionality of the data). On the contrary, the diffusion coordinates will capture the intrinsic organization of each data sets, and therefore will provide a canonical representation of the sets that can be used for matching the data. Note also that our approach does not require any prior information on the type of data we are dealing with.
- The other point that we wished to illustrate is the importance of using the density-invariant diffusion maps. As we will show, although the three sets have approximately the same intrinsic geometry (the data points lie on the same 2D submanifold), the distribution of the points on this manifold are quite different. Therefore, it is necessary to employ the density re-normalization technique described in Section II-B.

These two points constitute the main ingredients for a successful alignment of the sets.

We now describe the experiment in more details. Each set of frames was regarded as a collection of points in $\mathbb{R}^{10000}$, where the dimensionality coincides with the number of pixels per image. Following the lines of our algorithm, we formed a graph from each set with Gaussian weights $\exp(-\|x_i - x_j\|^2/\varepsilon)$. The quantity $\|x_i - x_j\|$ represents the L^2 norm between images i and j, and here again, the scale was chosen so that each data point would be numerically connected to at least one other data point. We expect each set to lie approximately on a manifold of dimension 2, as each subject essentially moved their head along two angles α and β shown on Figure 6 and as the light conditions were kept the same during the recording. Therefore, each data sets is the expression of a highly constrained mechanical system, namely the articulation between the neck and the head.

It is clear that the density of points on this manifold is essentially arbitrary and varies with each subject and recording. Indeed, the density is essentially a function of the type of movement of each subject, their speed of execution, and also the type of mask that they were wearing. Since we were only interested in the space of constraints, that is the geometry of the manifold, we renormalized the Gaussian weights according to the algorithm described in Section II-B, and constructed a Markov chain that approximates the Laplace-Beltrami diffusion. Figure 7 shows the embedding in the first three eigenfunctions for each data set. They are extremely similar. We then defined 8 matching triplets of landmarks in each set. The landmarks were chosen to correspond to the main head positions. We computed the diffusion embedding

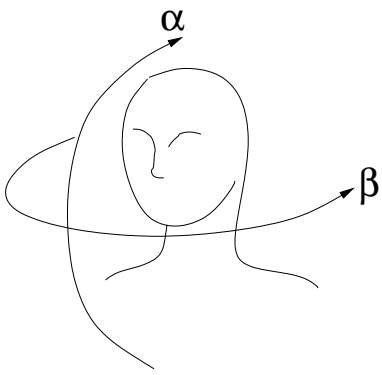
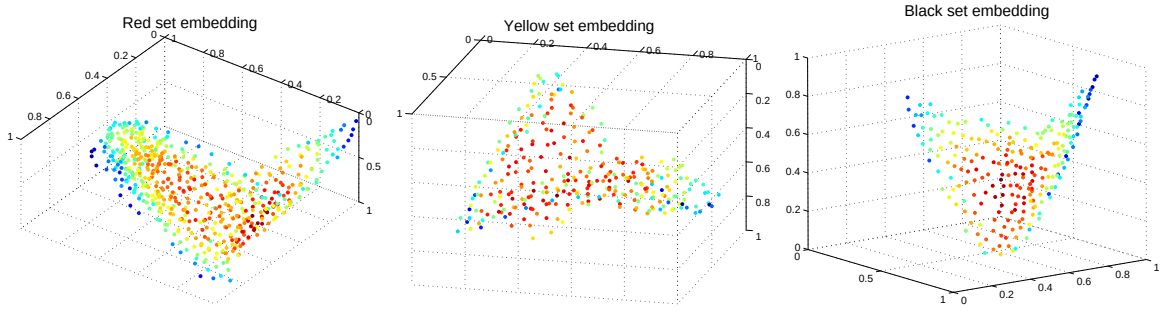


Fig. 6. Each subject essentially moved their head along the two angles α and β. There was almost no tilting of the head. Hence, the data points approximately lie on a submanifold of dimension 2.



(a) (b) (c)

Fig. 7. The embedding of each set in the first 3 diffusion coordinates. The color encodes the density of points. All three sets share this butterfly-shaped embedding

in 7 dimensions and we then calculated two affine functions $g_{YR} : \mathbb{R}^7 \to \mathbb{R}^7$ and $g_{YB} : \mathbb{R}^7 \to \mathbb{R}^7$ that match the landmarks from YELLOW to BLACK, and from YELLOW to RED.

Two conclusions can be drawn from this experiment. First, the diffusion embedding revealed that the data sets were approximately 2-dimensional, as expected (see Figure 7 for the embeddings in the first 3 diffusion coordinates). The diffusion coordinates captured the main parameters of variability, namely the angles α and β. From the embedding plots, it can be seen that all three embedded sets have strikingly similar shapes. *This supports our intuition that all sets should have similar intrinsic geometries*. From this observation, we were able to successfully compute two aligning functions g_{YB} and g_{YR}, and we used them to drive the movements of the black and red masks from those of the yellow mask. The result of the matching of the three data sets is shown on Figure 8. A live demo of this experiment can be found at [42].

The other conclusion concerns the importance of having used the density normalized diffusion coordinates. A key point in our analysis is that to compare the intrinsic geometries of each set, we need to be able to get rid of the influence of the points on the 2D submanifold. In order to underline the importance of this idea, we also computed the embedding of the three Yellow and BLACK without this renormalization. According to the discussion of Section II-B, the embedded sets should now reflect both the constraints (the intrinsic geometry) and the distribution of the points (the density on the submanifold). The result is shown on Figure 9, and although the embedding of the BLACK set still retain this butterfly shape that we previously obtained when renormalizing, the YELLOW set is now embedded as some portion of an ovoid. Although this statement can seem very qualitative, it is now clear that the alignment of these sets should fail. This experiment therefore underlines the importance of being able to compute density-invariant embeddings of the data.

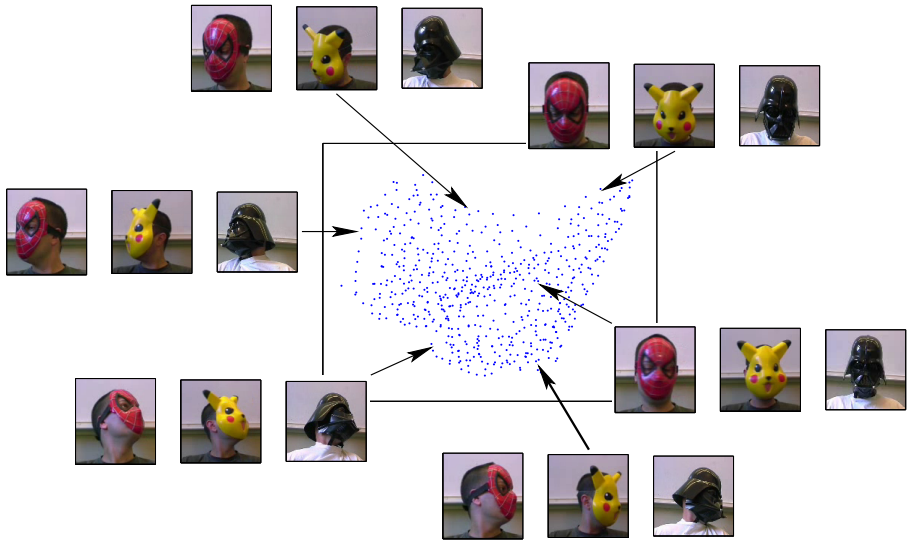
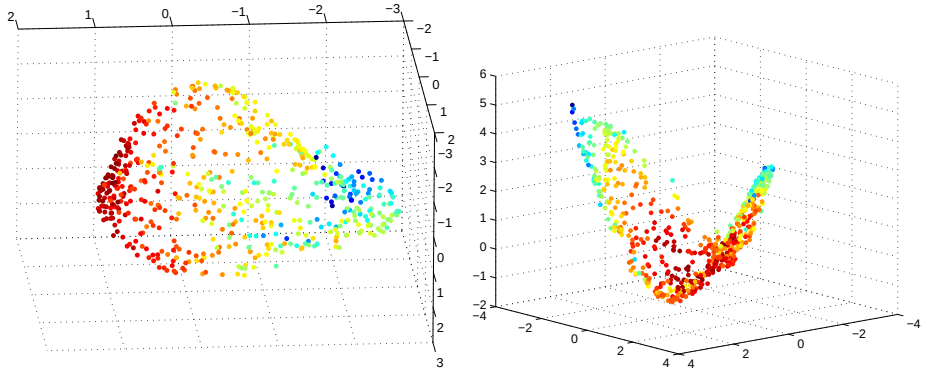


Fig. 8. The embedding of the YELLOW set in three diffusion coordinates and the various corresponding images after alignment of the RED and BLACK graphs to YELLOW.



(a) (b)

Fig. 9. The embeddings of the YELLOW (a) and BLACK (b) sets in three diffusion coordinates without the density renormalization. These embedded sets now have very different shapes, and their alignment is impossible.

IV. Conclusion and future work

In this work we introduced diffusion techniques as a framework for data fusion and multi-cue data matching by addressing several key issues. First, we underlined the importance of the Laplace-Beltrami normalization for data fusion by showing that it allows to merge data sets produced by the same source but with different densities. In particular, the Laplace-Beltrami embedding provides a canonical, density-invariant embedding which is essential for data matching. Second, we suggested a new data fusion scheme, by extending spectral embeddings using the geometric harmonics framework. Finally, we presented a novel spectral graph alignment approach to data fusion.

Our scheme was successfully applied to lip-reading where we achieved high accuracy with minimal preprocessing. We also demonstrated the alignment of high-dimensional visual data ("rotating heads" sequence).

In the work presented, we have focused on the situation when all sources are highly correlated. In the future we plan on extending our approach to multi-cue data analysis by integrating different signals from weakly correlated sources into a unified representation. This should open the door to applications related to multi-sensor integration. Finally, we also are studying a spectral based approach to the analysis

of signals as dynamical random processes. Our current work did not utilize the temporal information of the video sequences. By constructing a dynamical Markov process model, we intend to improve the lips reading accuracy.

V. Acknowledgments

The authors would like to thank Steven Zucker for helpful discussions during the course of this work and Andreas Glaser for helping us out for the data collection. We also express our thanks to the referees for their constructive questions and comments.

Appendix I
Existence and uniqueness of the stationary distribution

The goal of this section is to show that if the graph is connected, then the stationary distribution ϕ_0 is guaranteed to exist. The first step is to notice that the data set is finite, and therefore so is the state space of our Markov chain. Thus by a classical version of the Perron-Frobenius theorem, it suffices to prove that the chain is irreducible and aperiodic.

- The irreducibility is a mere consequence of the fact that the graph is connected. Indeed, let x_i and x_j be two data points, and let τ be the length of a path connecting x_i and x_j. Since the graph is connected, we know that $\tau < +\infty$. We conclude that $p_\tau(x_i, x_j) > 0$, which implies that the chain irreducible.
- Concerning the aperiodicity, remember that $w(\cdot, \cdot)$ represent the similarity between data points, so we can assume that for all data point x_i, we have $w(x_i, x_i) > 0$. Consequently, $p_1(x_i, x_i) > 0$, which implies that the chain is aperiodic.

Finally, we can conclude that our Markov chain has a unique stationary distribution ϕ_0.

Appendix II
Diffusion distance and eigenfunctions

The random walk constructed from a graph via the normalized graph Laplacian procedure yields a Markov matrix P with entries $p_1(x, y)$. As it is well known [15], this matrix is in fact conjugate to a symmetric matrix A with entries $a(x, y)$, given by

$$a(x,y) = \sqrt{\frac{d(x)}{d(y)}} p_1(x,y) = \frac{w(x,y)}{\sqrt{d(x)d(y)}} \,.$$

Therefore A has n eigenvalues $\lambda_0, ..., \lambda_{n-1}$ and orthonormal eigenvectors $v_0, ..., v_{n-1}$. In particular,

$$a(x,y) = \sum_{l=0}^{n-1} \lambda_l v_l(x) v_l(y) \,. \tag{9}$$

This implies that P has the same n eigenvalues. In addition, it has n left eigenvectors $\phi_0, ..., \phi_{n-1}$ and n right eigenvectors $\psi_0, ..., \psi_{n-1}$. Also, it can be checked that

$$\phi_l(y) = v_l(y) v_0(y) \text{ and } \psi_l(x) = v_l(x)/v_0(x) \,. \tag{10}$$

Furthermore, it can be verified that $v_0(x) = \sqrt{d(x)}/\sqrt{\sum_z d(z)}$, and therefore $\phi_0(y) = d(y)/\sum_z d(z)$ and $\psi_0(x) = 1$. In addition,

$$\phi_0(x)\psi_l(x) = \phi_l(x) \,. \tag{11}$$

It results from Equations 9 and 10 that P^t admits the following spectral decomposition:

$$p_t(x,y) = \sum_{l=0}^{n-1} \lambda_l^t \psi_l(x) \phi_l(y) \,, \tag{12}$$

together with the biorthogonality relation

$$\sum_{y\in\Omega}\phi_i(y)\psi_j(y)=\delta_{ij}\,, \tag{13}$$

where δ_{ij} is Kronecker symbol. Combining this last identity with Equation 11, one obtains

$$\sum_{y\in\Omega}\frac{\phi_i(y)\phi_j(y)}{\phi_0(y)}=\delta_{ij}\,.$$

This means that the system $\{\phi_l\}$ is orthonormal in $L^2(\Omega,1/\phi_0)$. Therefore, if one fixes x, Equation 12 can interpreted as the decomposition of the function $p_t(x,\cdot)$ over this system, where the coefficients of decomposition are $\{\lambda_l^t\psi_l(x)\}$.

Now by definition,

$$D_t(x,z)^2=\sum_{y\in\Omega}\frac{(p_t(x,y)-p_t(z,y))^2}{\phi_0(y)}=\|p_t(x,\cdot)-p_t(z,\cdot)\|^2_{L^2(\Omega,1/\phi_0)}\,.$$

Therefore,

$$D_t(x,y)^2=\sum_{l=0}^{n-1}\lambda_l^{2t}(\psi_l(x)-\psi_l(z))^2\,.$$

REFERENCES

[1] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.

[2] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 6, no. 15, pp. 1373–1396, June 2003.

[3] D. Donoho and C. Grimes, "Hessian eigenmaps: New locally linear embedding techniques for high-dimensional data," *Proceedings of the National Academy of Sciences*, vol. 100, no. 10, pp. 5591–5596, May 2003.

[4] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimension reduction via local tangent space alignement," Department of computer science and engineering, Pennsylvania State University, Tech. Rep. CSE-02-019, 2002.

[5] R. Coifman and S. Lafon, "Diffusion maps," *Applied and Computational Harmonic Analysis*, 2006, to appear.

[6] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: a geometric framework for learning from examples," University of Chicago, Tech. Rep. TR-2004-06, 2004.

[7] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the nyström method." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 214–225, 2004.

[8] Y. Bengio, J.-F. Paiement, and P. Vincent, "Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering," Université de Montréal, Tech. Rep. 1238, 2003.

[9] R. Coifman and S. Lafon, "Geometric harmonics: a novel tool for multiscale out-of-sample extension of empirical functions," *Applied and Computational Harmonic Analysis*, 2006, to appear.

[10] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps," *Proceedings of the National Academy of Sciences*, vol. 102, no. 21, pp. 7426–7431, May 2005.

[11] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker, "Geometric diffusions as a tool for harmonics analysis and structure definition of data: Multiscale methods," *Proceedings of the National Academy of Sciences*, vol. 102, no. 21, pp. 7432–7437, May 2005.

[12] B. Nadler, S. Lafon, R. Coifman, and I. Kevrekidis, "Diffusion maps, spectral clustering and the reaction coordinates of dynamical systems," *Applied and Computational Harmonic Analysis*, 2006, to appear.

[13] S. Lafon and A. B. Lee, "Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning and data set parameterization," *IEEE Pattern Analysis and Machine Intelligence*, 2006.

[14] R. I. Kondor and J. D. Lafferty, "Diffusion kernels on graphs and other discrete input spaces," in *ICML '02: Proceedings of the Nineteenth International Conference on Machine Learning*, 2002, pp. 315–322.

[15] F. Chung, *Spectral graph theory*. CBMS-AMS, May 1997, no. 92.

[16] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Tran PAMI*, vol. 22, no. 8, pp. 888–905, 2000.

[17] Y. Weiss, "Segmentation using eigenvectors: A unifying view." in *ICCV*, 1999, pp. 975–982.

[18] M. Meila and J. Shi, "A random walk's view of spectral segmentation," *AI and Statistics (AISTATS)*, 2001.

[19] S. X. Yu and J. Shi, "Multiclass spectral clustering." in *Proc. IEEE Int. Conf. Computer Vision*, 2003, pp. 313–319.

[20] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering." in *Advances in Neural Information Processing*, 2001, pp. 585–591.

[21] P. Diaconis and D. Stroock, "Geometric bounds for eigenvalues of markov chains," *The Annals of Applied Probability*, vol. 1, no. 1, pp. 36–61, 1991.

[22] M. Belkin and P. Niyogi, "Towards a theoretical foundation for laplacian-based manifold methods." in *COLT*, 2005, pp. 486–500.
[23] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in C*. Cambridge University, 1988.
[24] M. Gori, M. Maggini, and L. Sarti, "Exact and approximate graph matching using random walks." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 7, pp. 1100–1111, 2005.
[25] J. Ham, D. Lee, and L. Saul, "Semisupervised alignment of manifolds," in *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics, Jan 6-8, 2005, Savannah Hotel, Barbados*, pp. 120–127.
[26] X. Bai, H. Yu, and E. R. Hancock, "Graph matching using spectral embedding and alignment." in *ICPR (3)*, 2004, pp. 398–401.
[27] Y. Keselman, A. Shokoufandeh, M. F. Demirci, and S. J. Dickinson, "Many-to-many graph matching via metric embedding." in *CVPR (1)*, 2003, pp. 850–857.
[28] A. W. Fitzgibbon, "Robust registration of 2d and 3d point sets," in *Proceedings of the British Machine Vision Conference*, 2001, pp. 662–670.
[29] H. J. Wolfson and I. Rigoutsos, "Geometric hashing: An overview," *IEEE Comput. Sci. Eng.*, vol. 4, no. 4, pp. 10–21, 1997.
[30] M. Aharon and R. Kimmel, "Representation analysis and synthesis of lip images using dimensionality reduction," *Accepted to the International Journal of Computer Vision*.
[31] B. Christoph, C. Michele, and S. Malcolm, "Video rewrite: driving visual speech with audio," in *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1997, pp. 353–360.
[32] E. Cosatto and H. P. Graf, "Photo-realistic talking-heads from image samples." *IEEE Transactions on Multimedia*, vol. 2, no. 3, pp. 152–163, 2000.
[33] C. Bregler, S. Manke, and H. Hild, "Improving connected letter recognition by lipreading," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Processing*, 1993.
[34] N. Dettmer and M. Shah, "Visually recognizing speech using eigensequences," *Computational Imaging and Vision*, pp. 345–371, 1997.
[35] I. Matthews, T. Cootes, A. Bangham, S. Cox, and R. Harvey, "Extraction of visual features for lipreading." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 198–213, 2002.
[36] A. V. Nefian, L. H. Liang, X. X. Liu, X. Pi, and K. Murphy, "Dynamic bayesian networks for audio-visual speech recognition," *Journal of Applied Signal Processing*,, vol. 2002, no. 11,, pp. 1274–1288, 2002.
[37] J. Luettin, N. A. Thacker, and S. W. Beet, "Active shape models for visual speech feature extraction," in *Speechreading by Humans and Machines*, ser. NATO ASI Series, Series F: Computer and Systems Sciences, D. G. Storck and M. E. H. (editors), Eds. Berlin: Springer Verlag, 1996, vol. 150, pp. 383–390.
[38] J. Luettin and N. A. Thacker, "Speechreading using probabilistic models," *Computer Vision and Image Understanding*, vol. 65, no. 02, pp. 163–178, 1997.
[39] Y.-L. Tian, T. Kanade, and J. Cohn, "Robust lip tracking by combining shape, color and motion," in *Proceedings of the 4th Asian Conference on Computer Vision (ACCV'00)*, January 2000.
[40] I. Borg and P. Groenen, *Modern Multidimensional Scaling - Theory and Applications*. Springer-Verlag New York Inc., 1997.
[41] C. Bregler, S.Omohundro, M.Covell, M.Slaney, S.Ahmad, D.A.Forsyth, and J.A.Feldman, "Probabilistic models of verbal and body gestures," in *Computer Vision in Man-Machine Interfaces*, R. Cipolla and A. eds, Eds. Cambridge University Press, 1998.
[42] S. Lafon, "Demo of the mask alignment," 2005. [Online]. Available: http://www.math.yale.edu/∼sl349/demos_data/demos.htm.

Stéphane Lafon is a Software Engineer at Google. He received his B.Sc. degree in Computer Science from Ecole Polytechnique and his M.Sc. in Mathematics and Artificial Intelligence from Ecole Normale Supérieure de Cachan in France. He obtained his Ph.D. in Applied Mathematics at Yale University in 2004 and he was a research associate in the Mathematics Department during 2004-2005. He is currently with Google where his work focuses on the design, analysis and implementation of machine learning algorithms. His research interests are in data mining, machine learning and information retrieval.

Yosi Keller received the B.Sc. degree in electrical engineering in 1994 from The Technion-Israel Institute of Technology, Haifa. He received the M.Sc and Ph.D degree in electrical engineering from Tel-Aviv University, Tel-Aviv, in 1998 and 2003, respectively. From 1994 to 1998, he was an R&D Officer in the Israeli Intelligence Force. He is a visiting Assistant Professor with the Department of Mathematics, Yale University. His research interests include motion estimation and statistical pattern analysis.

Ronald R. Coifman is the Phillips Professor of Mathematics at Yale University. His research interests include: nonlinear Fourier analysis, wavelet theory, singular integrals, numerical analysis and scattering theory, and new mathematical tools for efficient computation and transcriptions of physical data, with applications to numerical analysis, feature extraction recognition and denoising. Professor Coifman, who earned his Ph.D. at the University of Geneva in 1965, is a member of the National Academy of Sciences and the American Academy of Arts and Sciences. He received the DARPA Sustained Excellence Award in 1996, the 1999 Pioneer Award from the International Society for Industrial and Applied Mathematics. He is a recipient of National Medal of Science.