Dense Non-rigid Surface Registration Using High-Order Graph Matching

Yun Zeng
Stony Brook University
yzeng@cs.sunysb.edu

Chaohui Wang
Laboratoire MAS, Ecole Centrale Paris
chaohui.wang@ecp.fr

Yang Wang
Carnegie Mellon University
wangy@cs.cmu.edu

Xianfeng Gu, Dimitris Samaras
Computer Science Department
Stony Brook University
{gu,samaras}@cs.sunysb.edu

Nikos Paragios
Laboratoire MAS, Ecole Centrale Paris, France
Equipe GALEN, INRIA Saclay - Ile-de-France
nikos.paragios@ecp.fr

Abstract

In this paper, we propose a high-order graph matching formulation to address non-rigid surface matching. The singleton terms capture the geometric and appearance similarities (e.g., curvature and texture) while the high-order terms model the intrinsic embedding energy. The novelty of this paper includes: 1) casting 3D surface registration into a graph matching problem that combines both geometric and appearance similarities and intrinsic embedding information, 2) the first implementation of high-order graph matching algorithm that solves a non-convex optimization problem, and 3) an efficient two-stage optimization approach to constrain the search space for dense surface registration. Our method is validated through a series of experiments demonstrating its accuracy and efficiency, notably in challenging cases of large and/or non-isometric deformations, or meshes that are partially occluded.

1. Introduction

3D surface registration is an important problem in computer vision with broad applications, such as 3D shape retrieval, face morphing, and object recognition [9]. It is particularly challenging when surfaces undergo large non-rigid deformations. In applications such as recognition of subtle facial expressions, there are localized, high-degree of freedom deformations. To tackle this problem, several approaches have been developed to obtain dense point correspondences by embedding the surfaces to a canonical domain which preserves the geodesics or angles [5, 6, 29, 32, 33]. Such embedding requires an initial set of feature correspondences or boundary conditions. Given noisy 3D scan data with varying scale, boundaries and resolutions, the performance of the above methods might suffer as it is difficult to find reliable feature point correspondences and consis-
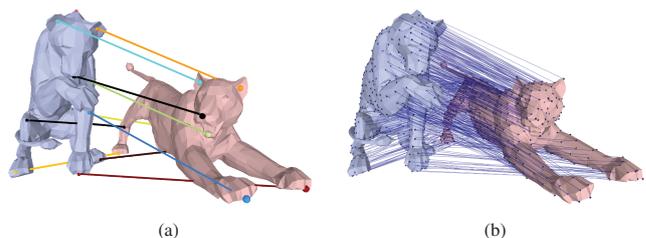
Figure 1. A dense matching result between two surfaces undergoing a large non-rigid deformation. Our approach can establish both the sparse (a) and dense (b) correspondences efficiently.

tent boundary conditions. To address this issue, in [34] a priority-driven strategy was considered to search for sparse feature correspondences based on the isometric assumption. In [22] a Möbius voting scheme was introduced to find correspondences between two sparse feature sets. However, both methods do not scale well to solve the dense registration problem. Furthermore, since most surface deformations are not perfectly isometric, solely considering intrinsic embedding information may introduce approximation errors to the matching result. Therefore, it is important to consider extrinsic similarity information as well to achieve accurate surface matching.

Recently it has been shown that graph matching is a powerful framework to establish feature correspondences, combining both appearance similarity and geometric compatibility [13, 28]. Previous work has applied graph matching to image features (*e.g.*, [21]). Considering only single matching scores, the matching problem becomes the well-known assignment problem [1]. For the pairwise matching problem, [28] proposed to use the dual-decomposition method which works well for a non-convex energy function. For the high-order matching problem, recent work, including probabilistic hypergraphs [31] and tensor matching [13], demonstrate good optimization results when the energy function is convex but the performance is unknown for non-convex energy functions.

Although graph matching has been successfully applied in 2D image features, limited prior art refers 3D surfaces. The obstacle lies in the fact that a 3D surface is not represented in the Euclidean 2D domain, and therefore distances between two points on the surface can not be computed in a closed form. For example, in [2, 26], only around 100 correspondences are found using geodesic information. The conformal mapping approach ([33, 29, 32]) is more flexible. According to the uniformization theory [14], any 3D surface can be conformally mapped to a 2D domain. However, such conformal mapping is not unique. The group of conformal mapping can be captured by the Möbius transform. In the case of mapping a surface that is topologically equivalent to a sphere, at least three correspondences are required to determine a unique conformal mapping. This transform can be computed in a closed form, which makes it possible for us to efficiently apply graph matching to surface registration. This motivates our new formulation of the 3D surface matching problem with the use of high-order graph matching that considers the cost of matching three correspondences. The main intuition is that the flexibility of this conformal mapping is represented by a Möbius transform. Since the Möbius transform is uniquely determined by fixing any three points on the surface, we can model the embedding energy effectively through the high-order graph interaction. Our paper is the first one applying high-order graph matching to surface registration.

In this paper, *we propose a new approach to achieve robust dense surface registration via high-order graph matching in the embedding manifold.* In particular, we use multiple measurements (*e.g.*, curvature, texture) to capture the appearance and geometric similarity between deformed surfaces and high-order graph interaction to model the implicit embedding energy. These measurements are used within a higher-order graph matching framework that is solved in an efficient manner by using a pseudo-boolean formulation [3]. Such an approach reduces the high-order term to quadratic terms [15] and obtains a near optimal solution based on the dual-decomposition technique [19]. Last, but not least a hierarchical algorithm is proposed to constrain the search space through candidate selection and local graph matching, which allows to achieve dense surface registration with a sub-vertex accuracy. As a result, our proposed method can establish *dense registration between non-rigid surfaces with large (potentially non-isometric) deformations, partial matching and inconsistent boundaries and scales.*

This paper is organized as follows. In Sec. 2, we formulate surface registration as a high-order graph matching problem. Sec. 3 presents our two-stage hierarchical surface matching framework. The implementation details and experimental results are presented in Sec. 4. Finally, we conclude our work in Sec. 5.

2. Mathematical formulation

As discussed in Sec. 1, a main challenge of dense surface matching is to capture large, high-degree-of-freedom deformations accurately. Instead of assuming the deformations near-rigid or near-isometric, we formulate surface registration as a high-order graph matching problem combining both extrinsic similarity and intrinsic embedding information. The dense surface matching is achieved by globally optimizing a functional which includes the cost of the deformation as well as the cost of correspondences according to multiple cues. Compared to existing approaches, our method can handle non-isometric surface deformations with partial overlapping. We formulate surface registration as a high-order graph matching problem combining both extrinsic similarity and intrinsic embedding information. Since this high-order graph matching has a non-convex energy function, it is in general difficult to solve directly using existing techniques such as [13]. In our proposed algorithm, a pseudo-boolean formulation is used to reduce the high-order term to quadratic terms [3]. Thus a globally optimal or near optimal solution can be obtained based on the dual-decomposition technique [19].

2.1. A pseudo-boolean formulation of the high-order graph matching problem

Let us denote by P_1 and P_2 the set of points from two surfaces S_1 and S_2 respectively. $P \triangleq P_1 \times P_2$ denotes the set of possible correspondences. We define the boolean indicator variable

$$x_a = \begin{cases} 1 & \text{if } a = (i,j) \in P \text{ is a correspondence,} \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

A basic constraint is that each point in P_1 is mapped to at most one point in P_2, while for each point in P_2 there is at most one point in P_1 mapping to it. Therefore, we have the set of feasible solutions defined as,

$$\mathcal{C} = \{\mathbf{x} \in \{0,1\}^{P_1 \times P_2} | \sum_{i \in P_1} x_{i,j} \leq 1, \sum_{j \in P_2} x_{i,j} \leq 1,$$
$$\forall i \in P_1 \text{ and } \forall j \in P_2\}. \quad (2)$$

Moreover as shown in [13], in order for matching to be scale invariant a high-order (degree 3) graph matching is required as follows

$$\min_{\mathbf{x} \in \mathcal{C}} \{ E(\mathbf{x}|\theta) = \sum_{a \in P} \theta_a x_a + \sum_{(a,b) \in P \times P} \theta_{ab} x_a x_b +$$
$$\sum_{(a,b,c) \in P \times P \times P} \theta_{abc} x_a x_b x_c \}, \quad (3)$$

where θ_a is the matching cost for each correspondence $a \in P$, θ_{ab} for a pair of correspondences $(a,b) \in P \times P$, and θ_{abc} for a triplet of correspondences $(a,b,c) \in P \times P \times P$.

Because the matching constraint 2 makes the optimization problem 3 difficult to solve, most existing works attempted to relax it (*e.g.*, [13]). In fact, the matching constraint can be reduced to pairwise terms in the energy function. We observe the following equivalence:

$$\forall i \in P_1, \sum_{j \in P_2} x_{i,j} \leq 1$$

$$\text{iff } \min_{x_{i,j}} \sum_{j',j'' \in P_2, j' \neq j''} \theta^{\infty} x_{i,j'} x_{i,j''} = 0 \quad (4)$$

where θ^{∞} is a sufficiently large number. We use $P^{\mathcal{C}}$ to denote the set of pairs that encodes the matching constraints for all the correspondences. Thus, the general high-order matching problem can be formulated as the following pseudo-boolean optimization problem [3]

$$\min_{\mathbf{x} \in \{0,1\}^{P_1 \times P_2}} \{E(\mathbf{x}|\theta) = \sum_{a \in P} \theta_a x_a + \sum_{(a,b) \in P \times P} \theta_{ab} x_a x_b + \sum_{(a,b) \in P^{\mathcal{C}}} \theta^{\infty} x_a x_b + \sum_{(a,b,c) \in P \times P \times P} \theta_{abc} x_a x_b x_c \}. \quad (5)$$

The above formulation is general and therefore is able to capture almost all matching scenarios (*e.g.*, partial matching) by properly defining the potentials.

Because of the positive weight θ^{∞} that encodes the matching constraint, the energy function 5 is nonconvex [4], and in general this is an NP-hard problem [3]. The advantage of the pseudo-boolean formulation is that theoretically any high-order terms can be reduced into a quadratic term [3] which can be done efficiently [15]. In this paper we employ the flexible dual-decomposition technique [19] which often obtains a near optimal solution. The details of the optimization algorithm is given in Sec. 2.3.

2.2. Potential functions

To consider multiple sources of similarity measurements, the potential functions in Eq. 3 are defined using both appearance and geometric information. For simplicity, we only consider the first and third order terms in our algorithm. Please note that the pairwise potential can also help improve the matching accuracy. For example, in the case of matching two surfaces with the same scale, we can consider geodesics on the surface. Furthermore, the potential functions can also be learned from a training set [8].

2.2.1   The singleton potential

For each correspondence (i, j) we consider both the geometric and texture (if available) information to define its potential as in [27]. For simplicity, we use the Gaussian curvature $\text{curv}(i)$ at point i, which is invariant under an isometric transformation [12], and the texture value $\text{tex}(i)$ at point i. The singleton potential for a correspondence (i, j) is defined as

$$\theta_{i,j} = (\text{curv}(i) - \text{curv}(j))^2 + \lambda_0 (\text{tex}(i) - \text{tex}(j))^2 \quad (6)$$

where λ_0 balances the weight between the curvature and the texture information. Similarly, other features can also be considered such as spin-image [16].

2.2.2   The high-order potential

According to the uniformization theorem [14], any 3D surface can be flattened conformally to a canonical 2D domain. Within such a mapping each feature point p has a parametric coordinate in the complex plane $z_p \in \hat{\mathbb{C}}$. The flexibility of this conformal mapping is represented by a Möbius transform, which can be uniquely determined by fixing any three points on the surface. Inspired by Lipman *et al.* [22], we compute the matching score between two triplets as the deformation error based on the Möbius transform.

Given two surfaces, S_1 and S_2, for any two triplets, $(p_i^1, p_j^1, p_k^1) \in S_1$ and $(p_i^2, p_j^2, p_k^2) \in S_2$, we first recover the associated Möbius transforms $m^1(z)$ and $m^2(z)$ that map each triplet to a constant configuration $(e^{i\frac{2\pi}{3}}, e^{i\frac{4\pi}{3}}, e^{i2\pi})$. On this new 2D domain, we find the mutually closest point correspondences set, denoted as

$$\mathcal{M}_{ijk} = \{z_p \rightarrow c(z_p) | p \in S_1, z_p, c(z_p) \in \hat{\mathbb{C}}\}, \quad (7)$$

and define the deformation error as

$$E(\mathcal{M}_{ijk}) = \sum_{l \in \mathcal{M}_{ijk}} |z_l - c(z_l)|^2. \quad (8)$$

Then we define the Möbius matching potential as follows,

$$\theta_{ijk}^{\text{Möbius}} = \begin{cases} \frac{E(\mathcal{M}_{ijk})}{|\mathcal{M}_{ijk}|^2} - 1. & \text{if } \frac{E(\mathcal{M}_{ijk})}{|\mathcal{M}_{ijk}|} < \delta \\ 1/|\mathcal{M}_{ijk}| & \text{otherwise} \end{cases} \quad (9)$$

Here δ is lower bound value to single out unlikely correspondences (in our experiment $\delta = 0.1$). Without it the minimization problem of Eq. 3 would encourage as many as possible correspondences even when some of them do not match. Intuitively, if there were more matching pairs and the distances between those matching pairs were smaller, the potential would be lower.

However, considering the Möbius energy alone can introduce certain ambiguity since it encodes only isometric information (an example is shown in Fig. 2). To resolve such ambiguity, we consider the Gaussian map of the surface. The Gaussian map is defined as the mapping of the normal at each point on the surface to the unit sphere [12]. The Gaussian map captures the extrinsic geometric information of the surface. In order to avoid ambiguities in orientation, the orientation of the Gaussian maps is considered
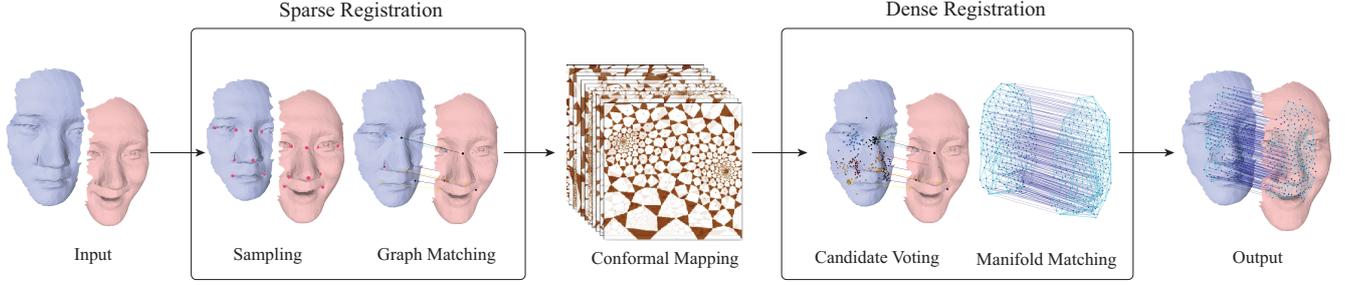
Figure 3. The outline of our algorithmic framework for surface matching as described in Sec. 3.
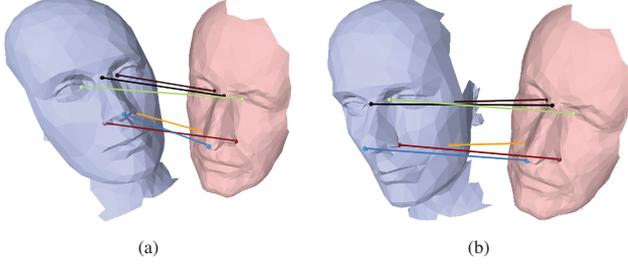
Figure 2. An example shows the ambiguity by considering only the intrinsic embedding information. The matching scores in (a) and (b) are the same from Eq. 8 based on the Möbius transform, since the distances between the matching features are identical. However, such ambiguity can be avoided by adding the extrinsic similarity information (*e.g.*, normal and curvature).

for each of the triplets. Two triplets have the same orientation if and only if the determinant of their normals have the same sign. Therefore, we have

$$\theta_{ijk}^{\text{Gaussian}} = \begin{cases} 0 & \text{if } \det(\mathbf{n}_i^1, \mathbf{n}_j^1, \mathbf{n}_k^1)\det(\mathbf{n}_i^2, \mathbf{n}_j^2, \mathbf{n}_k^2) \geq 0 \\ 1/|\mathcal{M}_{ijk}| & \text{otherwise} \end{cases}$$
(10)

Here $\mathbf{n}_i \in \mathbb{R}^3$ denotes the normal at point i, and $\det(\mathbf{n}_i, \mathbf{n}_j, \mathbf{n}_k)$ denotes the determinant of the 3×3 matrix $[\mathbf{n}_i, \mathbf{n}_j, \mathbf{n}_k]$. This is introduced as a soft constraint in our framework, because in the extreme case, the normal can reverse its orientations when the surface undergoes very large deformations. Finally, the triple potential for each possible triple matching $(p_i^1, p_j^1, p_k^1) \rightarrow (p_i^2, p_j^2, p_k^2)$ can be defined as

$$\theta_{ijk} = \lambda_1 \theta_{ijk}^{\text{Möbius}} + \lambda_2 \theta_{ijk}^{\text{Gaussian}}$$
(11)

2.3. Optimization and computational complexity

The idea of dual-decomposition is to re-formulate the original problem as the union of several sub-problems that are easier to solve [19, 18]. For the graph matching problem in Eq. 3, let θ denote the vector of the weights of the singleton, pairwise and triplet terms, and I denote the set of subproblems. The decomposition is represented by $E(\mathbf{x}|\theta) = \sum_{\sigma \in I} \rho_\sigma E^\sigma(x|\theta^\sigma)$, where ρ_σ is the weight for each subproblem. Then the original problem is solved by

updating the parameter θ^σ of each subproblem σ that increases the energy of the dual problem. Moreover, we have the following decomposition constraint:

$$\sum_{\sigma \in I} \rho_\sigma \theta^\sigma = \theta.$$
(12)

If we can find a lower bound $\Phi_\sigma(\theta^\sigma)$ for each subproblem, *i.e.*, $\Phi_\sigma(\theta^\sigma) \leq \min_x E^\sigma(x|\theta^\sigma)$, then we can obtain a lower bound for the original problem, *i.e.*,

$$\Phi(\theta) = \sum_{\sigma \in I} \rho_\sigma \Phi_\sigma(\theta^\sigma) \leq \sum_{\sigma \in I} \rho_\sigma E^\sigma(\mathbf{x}|\theta^\sigma) = E(\mathbf{x}|\theta). \quad (13)$$

In particular, we decompose the original problem into the following three subproblems:

1. a **linear subproblem** which considers only the singleton term $\sum_{a \in P} \theta_a x_a$. This linear subproblem is also known as the linear assignment problem [1].

2. a **higher-order pseudo-boolean subproblem** by reducing the high-order terms in 3 to quadratic terms [3] which can be solved by the QPBO algorithm [17]. Here we employ [15] for the reduction.

3. a **local subproblem** which divides the original surface into small regions and uses an exhaustive search to find the optimal solution in each small surface region.

Opposite to [28], a high-order pseudo-boolean subproblem is introduced because of the high-order terms in Eq. 3. After solving the subproblems, the dual variables $\{\theta^\sigma\}$ are updated by projecting to the space that satisfies Eq. 12 as in [19].

For the dual-decomposition algorithm above, the most expensive step in each iteration is the max-flow computation. In our paper, we use the popular implementation in [20], whose worst case complexity is $O(mn^2|C|)$, where m is the number of edges, n is the number of vertices, and $|C|$ is the cost of minimum cut. Assuming we select $|P_1|$ and $|P_2|$ feature points from two surfaces, there are $O(|P_1|^3|P_2|^3)$ possible triplets, each represented by a high-order term in Eq. 3. After the reduction from the high-order terms to quadratic terms, we can significantly reduce the complexity without searching for all possible matching correspondences.

3. Dense surface matching

The number of vertices n considered in this high-order formulation is the main computational bottleneck of our approach. In particular, when n becomes large, as in the case of dense surface registration, it is computationally expensive to solve the high-order graph matching problem. Furthermore, the optimality properties of the obtained solution degrade since the assumption of isometry is only an approximation and applies only when the features are far apart. The graph structure of the above matching problem would also be very complex if we consider all possible triplets. Several heuristic ways were considered to prune off some triplets, such as restricting the number of triangles per vertex [13]. However, because of the complexity of the problem, such pruning schemes often lead to erroneous matching results when the number of feature points is large. To reduce the computational complexity, we propose a two-stage optimization pipeline including *sparse feature matching* and *dense point matching*, as illustrated in Fig. 3.

In the **sparse feature matching** stage, an initial set of sparse feature points are selected among the local maxima of Gaussian curvature [22] on the input surfaces S_1 and S_2. Using our high-order graph matching algorithm in Sec. 2 we can compute the n_s correspondences between the two feature sets $\{p_1^1, p_2^1 \ldots, p_{n_s}^1\} \rightarrow \{p_1^2, p_2^2, \ldots p_{n_s}^2\}$. Extrinsic similarity and intrinsic embedding information are considered. p_i^1 and p_i^2 $(i = 1 \ldots n_s)$ are the matched feature points on S_1 and S_s, respectively. In this stage we only select a small set of feature points (typically $8 - 15$ in our experiments), so the computational cost is low on finding the sparse correspondences and computing the associated conformal maps.

Since the initial feature points are selected among the vertices and the middle points of the edges of the meshes, the matching results could be unreliable if the mesh resolution is low. To address the above issue, we consider all conformal maps induced by different Möbius transforms, which are determined by every three correspondences between two surfaces, for the **dense point matching**.

3.1. Candidate voting and clustering

Candidate voting From the sparse feature matching stage, we have a set of sparse correspondences $\{p_1^1, p_2^1 \ldots, p_{n_s}^1\} \rightarrow \{p_1^2, p_2^2, \ldots p_{n_s}^2\}$ between S_1 and S_2. Because the surface deformation might not be isometric, we propose a voting scheme based on Möbius transforms to compensate for the approximation error. Given any three correspondence pairs, $\{p_i^1, p_j^1, p_k^1\} \rightarrow \{p_i^2, p_j^2, p_k^2\}$, the Möbius transform can be computed efficiently in a closed form [22]. Under such a Möbius transform, any point $p^1 \in S_1$ will be mapped to a different candidate location $c(p^1) \in S_2$. Thus, for each point in the source surface, we

can compute the candidate locations in the target surface by considering all possible Möbius transforms from the feature correspondences. Please note that our candidate voting approach differs from the Möbius voting method [22] in two ways: 1) our method computes voting positions for each dense point rather than finding sparse feature correspondences and 2) multiple clusters are computed from the voting positions of each point and used to obtain a dense matching result.

One advantage of our voting approach is robustness. If any part of the sparse matching result from Sec. 3.1 is accurate, the matching candidates given by the Möbius groups will distribute closely around the true location. Another advantage is that this voting scheme provides a fast and effective way of constraining the search space for any point on the surface.

Candidate clustering Based on the above candidate locations, we want to use the underlying distribution to reduce our search space for the dense matching. It is also important that the dense matching should optimize the same objective as in the sparse matching stage. For any voting point $c(p^1) \in S_2$ of a source point $p^1 \in S_1$ that is obtained by aligning three correspondences $\{p_i^1, p_j^1, p_k^1\} \rightarrow \{p_i^2, p_j^2, p_k^2\}(i, j, k = 1 \ldots n)$, there is a cost $\theta_{ijk}^{\text{Möbius}}$ in the matching energy of Eq. 3. Intuitively, the lower the value of $\theta_{ijk}^{\text{Möbius}}$ and the closer the curvature and texture is, the more likely p^1 and $c(p^1)$ match. Therefore, we define the likelihood of each candidate matching $p^1 \rightarrow c(p^1)$ under the alignment of $\{p_i^1, p_j^1, p_k^1\} \rightarrow \{p_i^2, p_j^2, p_k^2\}$ as follows

$$f_{ijk}(p^1, c(p^1)) = e^{-\theta_{ijk}^{\text{Möbius}}} \qquad (14)$$

where $\theta_{ijk}^{\text{Möbius}}$ is the Möbius matching potential in Eq. 9. To obtain the candidate distribution for each point $p^1 \in S_1$, we use a kernel density estimate (KDE) with the density function defined as

$$\rho(p^1, c(p^1)) = \sum_c f_{ijk}(p^1, c(p^1)) K(\frac{\|c(p^1) - c(p_c^1)\|}{h}) \qquad (15)$$

where $c(p_c^1)$ is the center location of each kernel K in S_2 and h is the kernel bandwidth. The mean shift clustering [10] is employed to find the modes of this density. Compared to parametric representations, KDE has does not require nonlinear optimization to learn the distribution parameters.

Since we search for the modes in Eq. 15 on the 2D manifold instead of the 3D Euclidean embedding space, the distance function should be defined as the geodesic distance on the surface. However, as illustrated in Fig. 4 most of the candidate locations are close to the center, so the Euclidean distance is used in our method to simplify the mode search. To handle partial surface matching, we only select the modes with density higher than 0.1 and the closest point on the surface as the candidate matching point. If no such
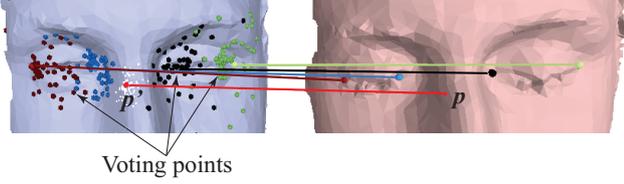
Voting points

Figure 4. Example showing voting points from different Möbius transforms and clustering. For any point p from the source surface, the clustering of its voting points on the target surface gives us a matching candidate p'.

mode exists, we report that there is no reliable matching point. The average number of resulting candidate points in our experiments is $1 - 6$. So our voting and clustering method can significantly reduce the search space.

3.2. Local high-order graph matching

Based on the matching candidates obtained for each vertex, our goal now is to find a good matching position locally for each dense point. This problem can be formulated similarly to the high-order graph matching problem defined in Sec. 2.1. Since the candidate voting scheme in Sec. 3.1 has removed the ambiguities caused by the Möbius transforms, we only need to consider the matching cost based on texture and geometric similarities defined in Eq. 6, as well as the orientation consistency defined in Eq. 10. Furthermore, the orientation consistency term can be defined locally, *i.e.*, each triangle $\triangle_{p_1 p_2 p_3}$ and its matched triangle $\triangle_{p_1' p_2' p_3'}$ should have the same orientation in the uniformization domain, which is known as having no flip in [24]. More specifically, for the three vertices of each triangle $\triangle_{123}$, we define the potential of matching (p_1, p_1'), (p_2, p_2') and (p_3, p_3') as follows

$$\theta_{123,1'2'3'} = \begin{cases} \theta^\infty & \text{sign}(\triangle_{123}) \neq \text{sign}(\triangle_{1'2'3'}) \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

Here θ^∞ is a sufficiently large number and $\text{sign}(\triangle_{123})$ and $\text{sign}(\triangle_{1'2'3'})$ denote the orientation of the triangle $p_1 p_2 p_3$ and $p_1' p_2' p_3'$, respectively, in the uniformization domain. From the candidate voting it is not guaranteed that every point has at least one matching candidate. Therefore, we remove the points without any matching candidate and obtain a triangulation for the remaining points on S_1 through the Delaunay triangulation algorithm [11] in the uniformization domain.

Suppose for each point $p \in S_1$, its matching candidates are given by $\mathcal{C}_p = \{p_i | p_i \in S_2, i = 1, 2, \ldots, n_p\}$. We define the boolean indicator variable:

$$x_p^i = \begin{cases} 1 & \text{if } p, p_i \in \mathcal{C}_p \text{ are correspondences} \\ 0 & \text{otherwise.} \end{cases} \quad (17)$$

Assuming that each $p \in S_1$ is matched to at most one of its candidates, we have the matching constraint:

$$\sum_{p_i \in \mathcal{C}_p} x_p^i \leq 1 \quad (18)$$

Therefore, the same optimization technique as described in Sec. 2.3 can be applied to solve the above problem.

Compared to the graph matching problem in Sec. 2.3, one major advantage of our local graph matching algorithm is that the number of matching candidates of each point is typically less than 6 and, therefore, the number of variables is very small. In particular, to match n points locally, there are only $O(n)$ variables and $O(n)$ triplet terms since the dense points are triangulated in the planar parametric domain.

4. Implementation and results

Our algorithm is implemented on an Intel® Xeon(TM) 3.4G PC with 4G RAM and an NVIDIA® Geforce 9800GTX+ graphics card. We developed a matching plugin for the open source software Meshlab[1]. For the mean shift algorithm, we used the source code available online[2]. For the potential functions of the graph matching algorithm defined in Sec. 2.2, the weights of Eq. 6 and 11 are defined as $\lambda_0 = 1$, $\lambda_1 = 0.1$ and $\lambda_2 = 1$, and the kernel bandwidth of Eq. 15 is set to be 0.01 times the diameter of the target surface.

The mid-edge uniformization algorithm was used for the conformal mapping [22, 23]. The computation of mid-edge uniformization involves solving a symmetric linear equation, which can be efficiently computed by GPU [7]. For a mesh with 10^4 faces the computation takes less than 1 second.

Since we consider almost all the triplets, the graph complexity scales cubically without pruning. Therefore, rather than searching for more sparse feature correspondences in the first stage, we try to find more accurate matching results for a few features. For example, 10 sparse feature correspondences will give us 120 voting positions for each point which are enough for finding candidate points. To match 10 feature points, the graph encoding step takes around 5 minutes and the graph matching step takes less than 1 minute. The voting and local high-order graph matching of 10^3 points based on the 10 sparse features takes around 1 minutes. Compared to previous work [22, 26] which only computes around 100 correspondences, our algorithm not only runs faster but also achieves more correspondences.

For the high-order graph matching algorithm in Sec. 2.3, the convergence of the dual-decomposition optimization depends on the input features. In our experiments, we observe that the more outliers (un-matched points), the more iterations it takes to converge.

---

[1]http://meshlab.sourceforge.net/
[2]http://www.caip.rutgers.edu/riul/research/code.html

4.1. Results

In this section we evaluate our new algorithmic framework for dense surface matching. In our experiments, we match surfaces with large deformations and inconsistent boundaries (partial overlapping). The number of vertices for each mesh is in the range of $1,500 - 4,000$. With our high-order graph matching algorithm, we can find the dense matching for $60 - 90$ percent of all vertices, which is illustrated as matched/total (no. of matched vertices/no. of total vertices of the source surface) for each example. The lion data of Fig. 1 comes from [25] and the face and hand data are captured with texture by the 3D scanner introduced in [30]. To measure the quality of dense registration, from the Delaunay triangulation of the points on the source surface, we consider the ratio of the area of each local triangle to the area of its matched triangle. For the natural deformations (*e.g.*, expression change, stretched arms or bending figures) we experimented with, the local area is not expected to undergo abrupt change. Therefore the area ratio is expected to be close to one for every local triangle.

Matching with largely inconsistent boundaries and partial overlapping: The mid-edge uniformization algorithm allows to map the boundaries of the surface to slits and preserve the conformal structure of the surface in an exact sense. Hence it is suitable for matching partially overlapping surfaces. This property can be combined with our candidate voting scheme to determine the outliers near the boundary where the mean shift clustering returns a low score. Examples are shown in Fig. 5, 6, and 7. An example of significant non-overlap between the two meshes is shown in Fig. 3.

Matching with large deformations: Fig. 7 and 8 show results that match two surfaces undergoing a large deformation. Even when the sparse features can not all be selected consistently (as shown in Fig. 8), our high-order graph matching algorithm in Sec. 2.3 is able to find reliable sparse correspondences (Fig. 8(a)) and obtain a dense surface matching result through the two-stage optimization scheme in Sec. 3(Fig. 8(b)).

Comparison experiments: Fig. 6 shows a comparison between our algorithm and the least square conformal mapping (LSCM) approach [29]. Although LSCM can handle free boundaries, there is no theoretical guarantee that the conformal structure is preserved near the boundary and it might include self-intersections in the mapping [24]. In our comparison, we use the feature correspondences computed from the sparse matching stage to initialize the LSCM experiments. The inaccuracy of the LSCM approach can be observed in Fig. 6(c). In this example, although all vertices on the left mesh are matched to the right mesh, there are approximately 42 percent flipped triangles. Note that here we cannot compare directly with the results in [29] where the initial feature points were manually selected.

(a) Sparse matching   (b) Dense matching

(c) Matching area ratio
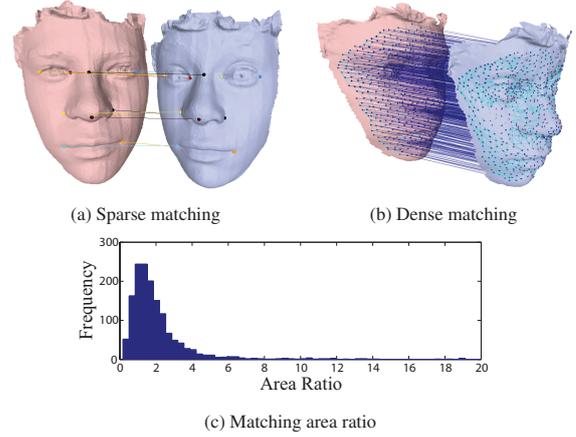
Figure 5. Example face matching result: (matched/total = 2098/2644)(best viewed in color)

(a) Sparse matching   (b) Dense matching

(c) Closeup of the dense matching.

(d) Matching area ration histogram
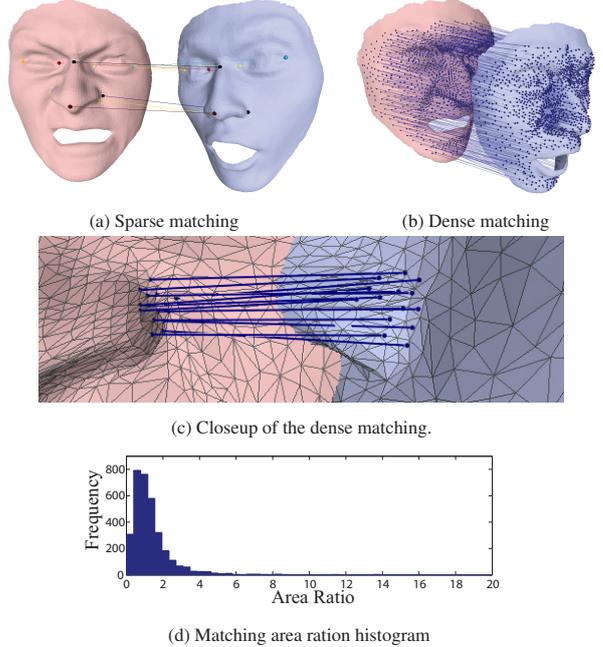
Figure 7. Dense matching under large non-rigid deformations. (matched/total = 2378/3633)(best viewed in color)

5. Conclusion

We proposed an algorithmic framework for non-rigid surface matching. In particular, a high-order graph matching formulation is used to combine the appearance and geometry similarity as well as the implicit embedding energy between deformed surfaces. Therefore, our proposed method can establish robust sparse registration between non-rigid surfaces with large deformations, partial matching and inconsistent boundaries and scales. Furthermore, a two-stage algorithm is proposed to constrain the search space through candidate selection and local graph matching, which allows to achieve dense surface registration with a sub-vertex accuracy. The method is modular both with respect to the density of points as well as the potentials used

(a) Sparse matching   (b) Dense matching   (c) LSCM matching   (d) LSCM error   (e) Our approach
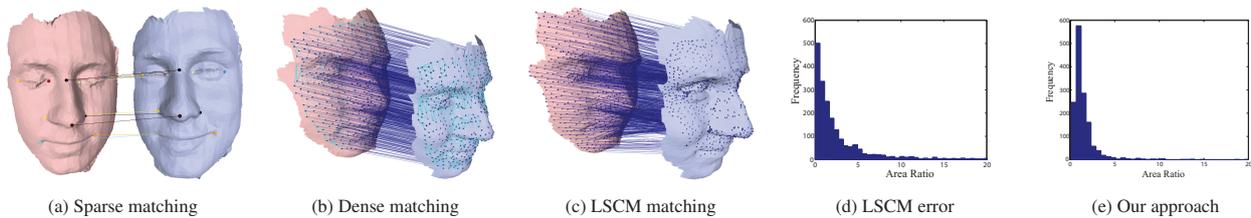
Figure 6. Comparison with LSCM approach [29] for dense surface matching. (matched/total = $1455/1635$) (best viewed in color). Notice the high number of flipped triangles in (c)

(a) Sparse matching   (b) Dense matching

(c)
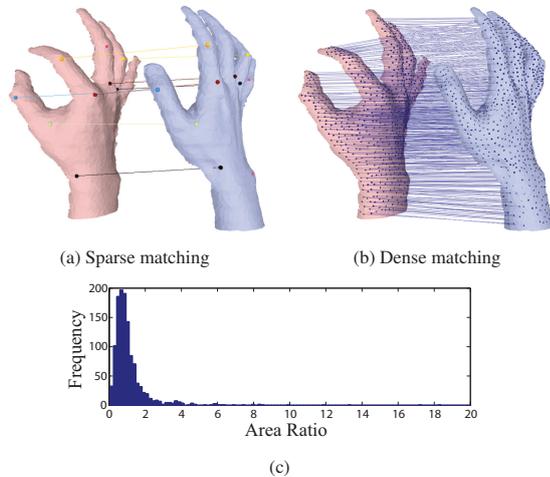
Figure 8. Dense matching under multiple articulated deformations. (matched/total = $1224/1786$)(best viewed in color)

to determine optimal partial correspondences.

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.

[2] D. Anguelov, P. Srinivasan, H.-C. Pang, D. Koller, S. Thrun, and J. Davis. The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces. In *NIPS*, 2004.

[3] E. Boros and P. L. Hammer. Pseudo-boolean optimization. *Discrete Applied Mathematics*, 123(1-3):155–225, 2002.

[4] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[5] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proc. National Academy of Sciences*, 103:1168–1172, 2006.

[6] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Expression-invariant representations of faces. *IEEE TPAMI*, 2004:1042–1053, 2007.

[7] L. Buatois, G. Caumon, and B. Levy. Concurrent number cruncher: a GPU implementation of a general sparse linear solver. *Int. J. Parallel Emerg. Distrib. Syst.*, 24(3):205–223, 2009.

[8] T. S. Caetano, J. J. McAuley, L. Cheng, Q. V. Le, and A. J. Smola. Learning graph matching. *IEEE TPAMI*, 31(6):1048–1058, 2009.

[9] R. J. Campbell and P. J. Flynn. A survey of free-form object representation and recognition techniques. *Comput. Vis. Image Underst.*, 81(2), 2001.

[10] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE TPAMI*, 24(5):603–619, 2002.

[11] M. de Berg, M. van Krefeld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer, 2 edition, 2000.

[12] M. P. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, 1976.

[13] O. Duchenne, F. Bach, I. Kweon, and J. Ponce. A tensor-based algorithm for high-order graph matching. In *CVPR*, 2009.

[14] H. M. Farkas and I. Kra. *Riemann Surfaces*. Springer, 2004.

[15] H. Ishikawa. Higher-order clique reduction in binary graph cut. In *CVPR*, 2009.

[16] A. Johnson. *Spin-Images: A Representation for 3-D Surface Matching*. PhD thesis, CMU, 1997.

[17] V. Kolmogorov and C. Rother. Minimizing nonsubmodular functions with graph cuts-a review. *IEEE TPAMI*, 29(7):1274–1279, 2007.

[18] N. Komodakis and N. Paragios. Beyond pairwise energies: Efficient optimization for higher-order MRFs. In *CVPR*, 2009.

[19] N. Komodakis, N. Paragios, and G. Tziritas. MRF optimization via dual decomposition: Message-passing revisited. In *ICCV*, 2007.

[20] N. Komodakis, G. Tziritas, and N. Paragios. Performance vs computational efficiency for optimizing single and dynamic MRFs: Setting the state of the art with primal-dual strategies. *Comput. Vis. Image Underst.*, 112(1):14–29, 2008.

[21] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *ICCV*, pages 1482–1489, 2005.

[22] Y. Lipman and T. Funkhouser. Möbius voting for surface correspondence. *ACM Trans. Graph.*, 28(3):1–12, 2009.

[23] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, 2(1):15–36, 1993.

[24] A. Sheffer, E. Praun, and K. Rose. Mesh parameterization methods and their applications. *Found. Trends. Comput. Graph. Vis.*, 2(2):105–171, 2006.

[25] R. W. Sumner and J. Popović. Deformation transfer for triangle meshes. In *SIGGRAPH*, pages 399–405, 2004.

[26] A. Tevs, M. Bokeloh, M. Wand, A. Schilling, and H.-P. Seidel. Isometric registration of ambiguous and partial data. In *CVPR*, 2009.

[27] N. Thorstensen and R. Keriven. Non-rigid shape matching using geometry and photometry. In *ACCV*, 2009.

[28] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization. In *ECCV*, 2008.

[29] S. Wang, Y. Wang, M. Jin, X. D. Gu, and D. Samaras. Conformal geometry and its applications on 3D shape matching, recognition, and stitching. *IEEE TPAMI*, 29(7):1209–1220, 2007.

[30] Y. Wang, M. Gupta, S. Zhang, S. Wang, X. Gu, D. Samaras, and P. Huang. High resolution tracking of non-rigid 3D motion of densely sampled data using harmonic maps. In *ICCV*, 2005.

[31] R. Zass and A. Shashua. Probabilistic graph and hypergraph matching. In *CVPR*, 2008.

[32] W. Zeng, Y. Zeng, Y. Wang, X. Yin, X. Gu, and D. Samaras. 3D non-rigid surface matching and registration based on holomorphic differentials. In *ECCV*, 2008.

[33] D. Zhang and M. Hebert. Harmonic maps and their applications in surface matching. In *CVPR*, 1999.

[34] H. Zhang, A. Sheffer, D. Cohen-Or, Q. Zhou, O. van Kaick, and A. Tagliasacchi. Deformation-driven shape correspondence. *Computer Graphics Forum (SGP)*, 27(5):1431–1439, 2008.