Volume xx (200y), Number z, pp. 1–14

Partial 3D Correspondence from Shape Extremities

Y. Sahillioğlu and Y. Yemez

Koç University, Computer Engineering Dept., Istanbul, Turkey

Abstract

We present a 3D correspondence method to match the geometric extremities of two shapes which are partially isometric. We consider the most general setting of the isometric partial shape correspondence problem, in which shapes to be matched may have multiple common parts at arbitrary scales as well as parts that are not similar. Our rank-and-vote-and-combine (RAVAC) algorithm identifies and ranks potentially correct matches by exploring the space of all possible partial maps between coarsely sampled extremities. The qualified top-ranked matchings are then subjected to a more detailed analysis at a denser resolution and assigned with confidence values that accumulate into a vote matrix. A minimum weight perfect matching algorithm is finally iterated to combine the accumulated votes into an optimal (partial) mapping between shape extremities, which can further be extended to a denser map. We test the performance of our method on several datasets and benchmarks in comparison with state of the art.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: 3D Shape Correspondence—partial shape correspondence, isometric distortion, extremity matching, partial isometry

1. Introduction

Finding correspondences between shapes is a fundamental problem in computer vision and graphics with numerous applications such as deformation transfer, statistical shape analysis, shape retrieval and registration [BBK08] [vKZHCO11]. The shape correspondence problem can be divided into two categories as complete and partial correspondence, where the latter deals with shapes that are common or similar only partially. Partial shape correspondence can also be thought of as a more general and hence harder variant of the former, since the partial matching set, which is a priori unknown, needs to be determined from the global set of surface points or mesh vertices that define a shape as a whole. In this paper, we address the partial correspondence problem, and consider it in its most general setting where shapes to be matched may have multiple common parts at arbitrary scales as well as parts that are not similar at all.

Isometry is an important clue in resolving shape correspondences since similar shape parts usually have similar metric structures. Although partial matching can be achieved by enforcing geodesic metric consistencies or by searching for partial mappings with minimum isometric distortion, the arbitrary scale of similar parts, which may change from one shape to the other, usually poses an important challenge that first needs to be resolved.

We propose a rank-vote-and-combine (RAVAC) algorithm to find correspondences between partially isometric shapes. We primarily target the partial correspondence problem, though the proposed scheme can be used to generate complete correspondences as well. Our algorithm collects partial isometry cues from the given shapes by considering all possible partial mappings (relations) between shape extremities and accumulates the collected information into a vote matrix which is then used to find an overall optimal partial correspondence via perfect graph matching. The main idea in RAVAC is to measure a correspondence pair's deviation from isometry based on only part of the shape. A small deviation from isometry gives a high confidence for that correspondence, and a large deviation gives a low confidence. Since the part segmentation is not available in advance, the algorithm computes an average deviation (distortion) value over many candidate segmentations. Each candidate segmentation is generated using a triplet of extremities from the source and target shapes. The "good" triplets needed to generate part segmentations are obtained by ranking all possible pairs of correspondences between extremities in advance and picking the triplets only from the pairs with low distortion estimates. To estimate distortions for ranking, we em-

ploy a heuristic based on pairs of k-tuples of extremities from the source and target with similar intrinsics.

The paper is organized as follows. In Section 2, we discuss the related work and elaborate on our contributions. Through Sections 3-7, we describe the main components of our correspondence scheme, which are sampling, *ranking*, *voting*, *combining* and dense matching, respectively. The computational complexity of the overall shape correspondence algorithm is relatively low, as analyzed in Section 8. We test the performance of our method on several datasets and benchmarks in comparison to two state of the art methods, as presented in Section 9, where we also discuss the limitations of our approach. We provide concluding remarks and possible directions for future research in Section 10.

We note that the source code and the executables for the method that we present in this paper are publicly available in http://home.ku.edu.tr/∼yyemez/partialcorresp.

2. Related Work

There are different ways of dealing with the scale problem in the literature, whether targeting partial or complete shape correspondence. Some methods simply assume that shapes come in compatible scales [GMGP05], [BBK06], [HAWG08], [TBW*09], [vKZH13] which is rather a strong assumption, whereas others normalize the original geometry with respect to some global intrinsic property such as maximum geodesic distance [SY11], [SY12a], [ZSCO*08], maximum centricity [ACOT*10] or total surface area [OMMG10], [PBB11]. Relying on global properties for normalization may lead to satisfactory results in the case of perfect isometry but may perform poorly when the shapes to be matched are nearly isometric. For partial matching on the other hand, the success depends highly on the degree of scale difference between similar parts of the shapes.

As a solution, some shape matching techniques rely on scale-invariant local shape descriptors [FS06], [ZBVH09], [BK10], [ZWW*10]. Local shape information is valuable for shape correspondence in the case of non-isometric deformations, but otherwise it is considered as less reliable than global shape information such as isometry. The methods which rely only on local geometric information may not perform well when the shapes to be matched exhibit large variations in their local geometry, or may easily confuse surface parts when there are many points that are locally similar. Hence some feature-based correspondence algorithms include also a pruning procedure that takes into account isometric clues by enforcing geodesic consistency [TBW*09], [ZSCO*08], [HAWG08], [ACOT*10]. Another important issue with the use of local shape descriptors, especially in the case of partial matching, is that different (uncommon) surface parts may interfere to computation of the descriptor at a given point. A very recent work [vKZH13] addresses this problem by introducing a local shape descriptor, namely the bilateral map, whose region of interest is defined by two feature points.

An alternative to geodesic metric for the measurement of isometric distortions is the diffusion metric which is less accurate but generally considered as more robust to topological noise [OMMG10]. Local scale differences are however difficult to handle using diffusion-based metrics. The commute-time metric for example addresses the scale problem only globally [WBBP11], hence cannot be used for the partial matching problem. Likewise, the heat kernel signature, as used in [PBB11], [DLL*10] to address the part matching problem, requires setting of a time scale parameter that itself depends on the global shape scale. A particular setting of the partial correspondence problem is part matching where one of the shapes to be matched is an isometric part of the other up to a scale [PBB11], [DLL*10], [SY12b], [KJS07]. In this setting, the correspondence-less approach in [PBB11] optimizes the region-wise similarity over the integration domains relying on diffusion-based local shape descriptors, whereas [SY12b] introduces a novel scale-invariant isometric distortion measure to address the scale normalization problem.

A common approach in the case of complete shape correspondence is to embed input shapes into spectral domain where the scaling problem is implicitly handled [JZ06], [MHK*08], [SY12a], [CH03]. These methods however treat the scale problem globally, hence cannot be applied to partial correspondence. A better alternative for partial matching is based on the Möbius transformation which is used for conformal embedding of the given shapes into a canonical coordinate frame on the complex plane where deviations from isometry are approximated based on mutually closest points [LF09]. This shape correspondence method is basically a voting technique (Möbius Voting), which aims to find a reliable but sparse matching between two partially isometric shapes. The algorithm iteratively samples a random triplet from each of the shape surfaces. The triplet pair then defines two Möbius transformations that embed the given shapes (after mid-edge flattening) into a canonical coordinate frame on the complex plane. Mutually closest points on this plane are considered as candidates for correspondence and voted based on the distances in between. The final output of the algorithm is a set of correspondences each associated with a confidence value. The Möbius Voting (MV) method is capable of producing a small number of reliable correspondences, but usually fails to achieve a reliable dense matching. Although good triplets of surface points can bring the accommodating parts of the given shapes to the same pose and scale successfully, the same transformation applied to other parts that do not necessarily expect the same transformation may easily distract the global voting process. The experiments conducted in [LF09] actually show that the method becomes unstable when the input shapes exhibit less than approximately 40% similarity.

Following [LF09], several methods that use Möbius transform for shape matching have then been proposed, though not in the context of partial correspondence [ZWW*10], [KLCF10], [KLF11], [LAAD11]. In particular, the Blended Intrinsic Maps (BIM) method of [KLF11] can be considered as an extension of MV, specifically designed to address the complete dense correspondence problem. Instead of a voting approach, the BIM method uses blending: It generates many complete maps between shapes via Möbius transform based on triplets of extremal points, weights these maps at every surface point by distortion and then blends them into a final map by computing an approximate geodesic centroid for every mapped point. BIM works very well in the case of complete shape matching, but does not support partial matching since it is essentially based on generation of *complete* candidate maps. Such complete maps do not actually exist when the shapes to be matched have dissimilar parts that constrain the distortion estimate. Theoretically one could envisage using BIM to find partial correspondences since it blends the generated complete maps by weighting. This would however yield robustness problems similarly as MV (in fact more severely than MV), as we will demonstrate by experiments in this paper. In contrast to these two methods, our method explicitly explores the space of *partial* maps defined over shape extremities. These partial maps are populated via region of interest sampling and used to accumulate partial isometric clues (distortions) into a vote matrix. Hence we use *voting* to match shape extremities and *blending* to extend the obtained sparse correspondence to a dense one. We note that, for the case of complete dense correspondence, the BIM method has been outperformed by several recent works based on functional representation of correspondences [OBCS*12], [PBB*13], [ROA*13], which however lack partial shape matching support.

Another state of the art correspondence method is the deformation-driven technique of [ZSCO*08], which can handle non-isometric shape variations (up to a certain degree) as well as partial isometries. In this method, an optimal correspondence is sought between shape extremities via priority-based combinatorial tree traversal by pruning the search space according to some criteria based on local shape similarity and geodesic consistency. For each candidate correspondence set, the source shape is deformed to the target based on these small number of landmarks (anchor points), and the correspondence with the smallest distortion gives the best matching. The major drawback of this scheme is the extensive computational load due to the process of repeated deformations. Another shortcoming is the need for error threshold parameters employed in tree pruning, which are usually data dependent. Hence it is often very difficult to set these parameters correctly and the combinatorial tree traversal may easily miss some of the correct feature pairings. Moreover, the geodesic information which is used to prune the combinatorial search tree is normalized based on some global instrinsics, which is problematic for matching arbitrarily scaled shape parts as discussed before.

2.1. Contributions and advantages

In our previous work [SY12b], we have addressed the part matching problem (not the partial correspondence problem in the most general setting) and described a method that also relies on shape extremities. However the framework described in that work is completely different than our current solution and actually very simplistic, aiming to introduce a novel scale-invariant isometric distortion measure. Its focus is on promoting this novel distortion measure, not a partial matching algorithm. The method simply assumes that the top M shape extremities of one shape are all included in the other shape as well and runs a combinatorial search over all possible permutations to match these extremities with M extremities of the other, minimizing the proposed novel distortion measure. The method that we present in this current paper does not use this distortion metric and does not either employ such a simplistic combinatorial search, rather it accumulates partial isometric clues by traversing all possible partial maps, employing more sophisticated algorithms for ranking, voting and combining.

There are few methods in the literature, that are capable of addressing the partial correspondence problem in the most general setting where shapes may have multiple common parts at arbitrary scales as well as parts that are not similar [LF09], [FS06], [TBW*09], [ACOT*10], [ZSCO*08]. All these methods mainly rely on scale-invariant local shape descriptors except for the MV method [LF09]. Note also that the methods in [TBW*09], [ACOT*10], [ZSCO*08] enforce geodesic consistency in addition to local shape similarity, and hence resort to global intrinsic properties for shape normalization. When compared to MV, our method has several advantages. First, we handle the scale problem inherent to partial correspondence directly in the 3D Euclidean space wherein isometry is originally defined, hence as free of embedding errors. Second, our method can produce reliable *dense* correspondences between partially isometric shapes. Third, we impose no restriction on shape topology. Last, our method generates more reliable and accurate correspondences, especially at shape extremities, and can handle shape pairs with less similarity overlap.

In view of the above discussion, the main contribution of this work is a computationally efficient and robust method that can accumulate partial isometric clues into a vote matrix and thereby computes partial shape correspondences which can be dense or sparse. We note that the focus of this work is on partial correspondence, though the proposed algorithm can also generate complete correspondences.

3. Sampling

We pick shape extremities of the given shapes by using local extrema of the integral geodesic distance function [HSKK01]. Let $\mu(v)$ denote the integral geodesic distance at vertex v. Prior to computation of μ, we apply Laplacian smoothing to each shape model to prevent samples at noisy

bumps. We then initialize the sample sets with local maxima and minima of μ. The local maxima are expected to be on the tips of a given shape whereas local minima correspond to surface points which lie near the center of the shape [ZSCO*08]. The initial sample sets are then exposed to two steps of pruning, first of which clusters geodesically close samples into the most extreme ones where the closeness threshold is determined based on the maximum geodesic distance $g_{\max}$ on the surface. In our experiments, we have used the value obtained by dividing $g_{\max}$ with a factor $h \in [10,20]$ depending on the dataset. The parameter h basically determines the scale of sampling, which we set manually by experimenting. The second step of pruning removes a local maximum (minimum) v from the sample set if $\mu(v)$ is less (greater) than the average μ to cancel out redundant extremities that are not on tips (central region). The vertices resulting from this sparse sampling process constitute the sets S (source) and T (target) to be matched (Fig. 1).

4. Ranking

In the ranking phase, we rank all possible pairs of correspondences between extremities based on their deviations from isometry. We estimate the deviation for each pair, hence the isometric distortion, using a heuristic based on pairs of k-tuples with similar average normalized geodesic distances on the source and target. We describe the ranking process in detail in the sequel (see also the pseudocode of the overall correspondence algorithm given at the end of this section).

4.1. Distortion estimate

Given a mapping $\S : S \to T$, i.e., a set of correspondence pairs, we measure the isometric distortion D_{iso} as follows:

$$D_{\text{iso}}(\S) = \frac{1}{|\S|} \sum_{(s_i,t_j)\in\S} d_{\text{iso}}(s_i,t_j,\S') \qquad (1)$$

where $d_{\text{iso}}(s_i,t_j,\S')$ is the contribution of the individual correspondence (s_i,t_j) to the overall isometric distortion:

$$d_{\text{iso}}(s_i,t_j,\S') = \frac{1}{|\S'|} \sum_{(s_l,t_m)\in\S'} |g(s_i,s_l) - g(t_j,t_m)| \qquad (2)$$

where $g(.,.)$ is the geodesic distance between two vertices on a given surface. The traversal list $\S'$, which is by default $\S - \{(s_i,t_j)\}$, includes the correspondence pairs to be traversed in order to compute the distortion of a given individual correspondence pair (s_i,t_j). Note that variants of the isometric distortion function defined by (1) can also be found in [BBK06] as well as in most of our previous work [SY11], [SY12a], [SY13].

An important issue in computation of the isometric distortion is how to normalize the scale of the geodesic distance function g involved in Eq. 2 since, in the case of partial matching, there are no agreed maximum geodesic distances on the source and target due to possible local scale differences. The key observation here is that the individual isometric distortion of a queried match (s_i,t_j) can safely be evaluated via Eq. 2 in the absence of globally normalized geodesics by using a traversal list consisting of matches from the shape part where the pair s_i and t_j itself resides in. The geodesic distances for this query can be normalized by using the maximum geodesic distance within this shape part. However since the corresponding shape parts are not known in advance, we estimate the individual isometric distortion by traversing over all possible one-to-one mappings of cardinalities 2 to 5. Note that these mappings do not include the query (s_i,t_j) and the cardinality of a mapping is defined as the number of pairs in it. We do not check beyond 5 due to efficiency reasons as well as the fact that 5 extremities (plus s_i or t_j) are usually sufficient to represent any given shape part, e. g., large-scale limbs in humans and animals. The estimate, $\tilde{d}_{\text{iso}}(s_i,t_j)$, of the individual isometric distortion of the correspondence (s_i,t_j) is then computed by

$$\tilde{d}_{\text{iso}}(s_i,t_j) = \frac{1}{4} \sum_{k\in[2,5]} \min_l \{d_{\text{iso}}(s_i,t_j,\S_l^{(k)})\} \qquad (3)$$

where $\{\S_l^{(k)} \mid l = 1,2,...,L_k\}$ is the set of all maps of size k, not including (s_i,t_j), and $L_k = \binom{|S|-1}{k}\binom{|T|-1}{k}(k!)$. We denote this set by $\mathcal{S}^{(k)}$. While computing the distortion via Eq. 3, the geodesic distance function g is normalized for each shape with the maximum geodesic distance between the k samples of the given mapping. Taking the minimum in (3) guarantees that if (s_i,t_j) is a good match and traverses a list of matches from the same shape part it resides in, then this is appreciated by selecting the lowest distortion. We then average over sets of maps with different cardinalities since maps of small size, e.g., with $k = 2$ or 3, are likely to fall in the same part as (s_i,t_j) but may exhibit symmetric flip problems, whereas mappings with large cardinalities, e.g., $k = 4$ or 5, are unlikely to be confused by flips but have the risk of including irrelevant samples from a distinct part.

4.2. Safe map generators

In Eq. 3, each (s_i,t_j) traverses all possible one-to-one mappings to compute the minimum distortion over $\mathcal{S}^{(k)}$. To reduce computation, we prune $\mathcal{S}^{(k)}$ so as to keep only the *potentially safe* maps, i.e., the maps between k source samples and k target samples which are expected to be from similar shape parts (see Fig. 1).

To this end, for each k, we define a set of safe map generators, $\mathcal{G}^{(k)}$, which contains all pairs of k-tuples, one tuple from the source sample set and the other from the target, such that any map between these tuples is potentially safe. We denote each of these pairs of sample tuples by $G_m^{(k)} \in \mathcal{G}^{(k)}$ for $m \in [1,|\mathcal{G}^{(k)}|]$. A pair of k-tuples is identified as a safe map generator if it satisfies the geodesic consistency condition that the average of pairwise normalized geodesics between source samples is close to that of between target samples. We normalize the geodesics with the maximum geodesic distance between the samples of the given tuple. Note that

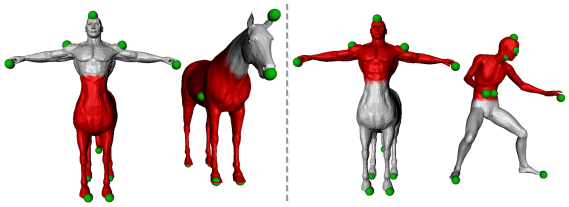

Figure 1: *Samples (green spheres) on two different shape pairs. A safe map generator $G_m^{(k)}$ is expected to choose k samples from each common shape part (red regions).*

although all $k!$ mappings generated from a given $G_m^{(k)}$ are referred to as potentially safe, only a small portion of them are actually correct mappings between two similar parts. Hence while evaluating a query match (s_i,t_j) via Eq. 3, taking the minimum helps eliminating the contribution of the irrelevant partial maps.

We create the generator sets $\mathcal{G}^{(k)}$ incrementally for $k = 3,4,5$ (no pruning is applicable for $k = 2$). For $k = 3$, each triplet of source samples is tested with each triplet of target samples to meet the geodesic consistency condition. Among $\binom{|S|}{3}\binom{|T|}{3}$ pairwise triplet combinations, typically $20\% - 30\%$ make into $\mathcal{G}^{(3)}$ in our experiments, where the closeness threshold is manually set as 0.15 by experimenting. For $k = 4,5$, we incrementally build $\mathcal{G}^{(k)}$ from $\mathcal{G}^{(k-1)}$. In each case, a pair of source and target samples appended to an existing generator $G_m^{(k)}$ triggers a new geodesic consistency test and typically $2\% - 4\%$ of all possible pairwise combinations are selected. Some safe map generators from $\mathcal{G}^{(3)}$ are demonstrated in Fig. 2.

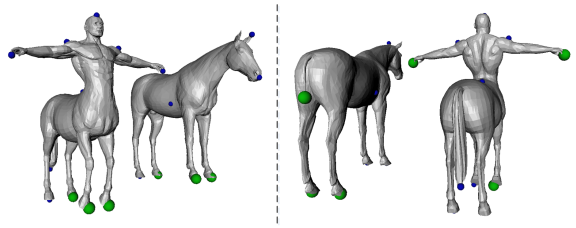

Figure 2: *Three different triplet pairs (safe map generators) from $\mathcal{G}^{(3)}$ are indicated with large green spheres on three different views of the same shape pair. Blue spheres represent the remaining samples.*

By replacing $\mathcal{S}^{(k)}$ in Eq. 3 with the potentially safe one-to-one maps based on $\mathcal{G}^{(k)}$, we not only reduce the search space significantly but also increase the accuracy by excluding unexpected distortion values. These unexpected high distortions are due to evaluation of (s_i,t_j) via (unsafe) maps that accommodates samples from irrelevant shape parts. Once the individual distortions are computed via Eq. 3, for each source sample s_i, we rank the pairs (s_i,t_j) based on

Input: Extremity sample sets S and T
Output: One-to-one mapping $\S^* : S \to T$
———— Ranking ————
$\mathcal{G}^{(k)}$ for $k = 3,4,5$: safe map generators, i.e., all pairs of k-tuples of extremities from the source and target with similar intrinsics;
For each $s_i \in S$
  Estimate $\tilde{d}_{\text{iso}}(s_i,t_j)\ \ \forall t_j \in T$ based on $\{\mathcal{G}^{(k)}\}$ via (3);
  Qualify the match (s_i,t_k) for voting if $d_{\text{iso}}(s_i,t_k)$ appears before the first significant jump in the sorted distortion plot of s_i;
———— Voting ————
Γ: Vote matrix with all entries γ_{ij} initialized to 0;
For $m = 1$ to $|\mathcal{G}^{(3)}|$
  If $G_m^{(3)} = ((s_{i_1},s_{i_2},s_{i_3}),(t_{j_1},t_{j_2},t_{j_3})) \in \mathcal{G}^{(3)}$ generates
  $\S_l^{(3)} = \{(s_{i_1},t_{j_1}),(s_{i_2},t_{j_2}),(s_{i_3},t_{j_3})\}$ where all pairs are qualified
    Bring meshes to the same scale by multiplying target with $\kappa = (\frac{g(s_{i_1},s_{i_2})}{g(t_{j_1},t_{j_2})} + \frac{g(s_{i_1},s_{i_3})}{g(t_{j_1},t_{j_3})} + \frac{g(s_{i_2},s_{i_3})}{g(t_{j_2},t_{j_3})})/3$;
    Set $S_l = \{s_{i_1},s_{i_2},s_{i_3}\}$ and $T_l = \{t_{j_1},t_{j_2},t_{j_3}\}$;
    Compute regions of interest, S_l and T_l, on source and target;
    Spread $\sim$100 dense samples, $\hat{S}_l$ and $\hat{T}_l$, on regions of interest;
    Find the dense map $\hat{\S}_l : \hat{S}_l \to \hat{T}_l$;
    Vote up confidence of extremity match $(s_i,t_j) \in \S_l^{(3)}$ via $\gamma_{ij} = \gamma_{ij} + \exp(-d_{\text{iso}}(s_i,t_j,\hat{\S}_l))$;
———— Combining ————
Set the cost matrix $\mathbf{C}^* = \infty$;
$c_{ij}^* = 1 - \gamma_{ij}$ for high-confidence matches (s_i,t_j);
Repeat
  $\S^* =$ minimum-weight perfect matching on $\mathbf{C}^*$
  Let (s_a,t_b) be the least-confident match in $\S^*$;
  $c_{ab}^* = \infty$;
Until there is no jump in confidences of the matches in $\S^*$

Figure 3: *The overall RAVAC Algorithm.*

their individual distortions: We sort all possible $|T|$ different matches with respect to $\tilde{d}_{\text{iso}}(s_i,t_j)$ in ascending order and qualify only the ones with a distortion value that appears before the first significant jump in the corresponding distortion plot. We assume that a significant jump occurs where the difference between two consecutive values becomes larger than the sum of the first two distortion differences, i.e., the sum of the difference between the first and the second values, and the difference between the second and the third values in the sorted list. A similar jump thresholding heuristic is employed also in other shape correspondence works such as in [ZSCO*08] for determining the optimal feature size and in [SY13] for tracking symmetric flips. With the qualified matches, the voting module is then ready to start, as described next (see also the pseudocode in Fig. 3).

5. Voting

With the ranking of possible matches in hand, one possibility to solve the correspondence problem is to select the least distorted match for each source sample. This straightforward solution would give a (possibly many-to-one) mapping that would however suffer from symmetric flips and mismatches due to low number of extremities being matched. We therefore consult to a voting procedure which is more robust, that relies on the ranking obtained in the previous section. The

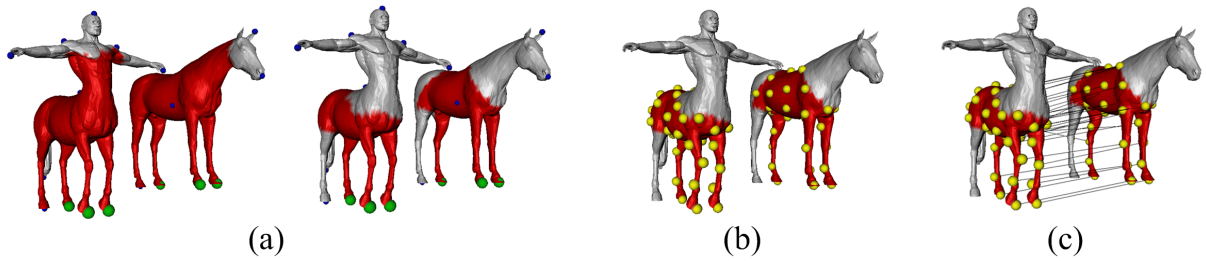

Figure 4: *An example of the voting process for a generating pair of sample triplets from $\mathcal{G}^{(3)}$. a) Two steps that decide regions of interest (painted red), b) evenly-spaced dense samples (yellow spheres), and c) one-to-one map between them (lines) to be used for computation of confidence votes.*

basic idea is as follows. We accumulate confidence votes for all possible pairs of correspondences between extremities into a vote matrix. These confidence votes are collected based on the isometric distortions of the pairs. The distortions are computed over many part segmentations generated using triplets of extremities on the source and target. Hence the voting process considers only the generator set $\mathcal{G}^{(3)}$ (the others are discarded simply due to computational reasons). Among 3! potentially safe maps generated from each $G_m^{(3)} \in \mathcal{G}^{(3)}$, only those containing the matches qualified in the ranking phase are taken into account. Each such safe map $\S_l^{(3)}$ defines two regions of interest, hence two part segmentations, on the given two shapes (as will be explained next), which are resampled and matched at a denser level (see Fig. 4). The resulting isometric distortion is then used to vote for the three matches contained in this potentially safe map. This is repeated for all qualified safe maps and the resulting votes are accumulated into a vote matrix where each entry represents the confidence of a potential match between two shape extremities. In the sequel, we describe the voting algorithm in detail.

5.1. Finding regions of interest

Let $\S_l^{(3)}$ be a potentially safe map generated from $G_m^{(3)} = ((s_{i_1}, s_{i_2}, s_{i_3}), (t_{j_1}, t_{j_2}, t_{j_3}))$ such that $\S_l^{(3)} = \{(s_{i_1}, t_{j_1}), (s_{i_2}, t_{j_2}), (s_{i_3}, t_{j_3})\}$. The voting algorithm first brings the shapes to the same scale by scaling the target mesh with a factor $\kappa = (\frac{g(s_{i_1}, s_{i_2})}{g(t_{j_1}, t_{j_2})} + \frac{g(s_{i_1}, s_{i_3})}{g(t_{j_1}, t_{j_3})} + \frac{g(s_{i_2}, s_{i_3})}{g(t_{j_2}, t_{j_3})})/3$ based on the geodesic distance ratios between the ordered sample points, and then finds the regions of interest that these shape extremities determine (see Fig. 4a). Let the extremity sample sets $\{s_{i_1}, s_{i_2}, s_{i_3}\}$ and $\{t_{j_1}, t_{j_2}, t_{j_3}\}$ be denoted by S_l and T_l, respectively. The region of interest on the source shape includes the source mesh vertices that are close to S_l and distant to $S - S_l$. To implement this, we mark a vertex v as a region vertex if $g(s,v) < g_{l,\max} \;\; \forall s \in S_l$, where $g_{l,\max}$ is the maximum geodesic between extremity samples in S_l (see Fig. 4a-left). To meet the second requirement, for each maximal extremity $s' \in S - S_l$, we unmark the region vertices that are at most $g(s,s')/2$ apart from s' where $s \in S_l$ is the closest extremity to s' (see Fig. 4a-right). The region of interest on the target shape defined by the extremity set T_l is computed likewise.

5.2. Dense region sampling

Next, we distribute evenly-spaced dense samples in the regions of interest (see Fig. 4b). We resample and populate the region of interest on the source shape by first selecting the corresponding extremities as the first three dense samples. Given the region area A, we use the ad-hoc formula to compute the radius $r = 0.17\sqrt{A/\pi}$ that ensures evenly sampling of about 100 dense samples [SY11]. The sampling procedure is as follows. When an arbitrary region vertex is selected as a dense sample, all the region vertices lying within its patch of radius r are marked. The next dense sample is then selected arbitrarily from the unmarked region vertices. When this is repeated until no unmarked region vertex is left, we obtain a partitioning of the region into dense samples that are at least r apart from each other [HSKK01]. A similar evenly-spaced sampling on the regions of interest of the scaled target mesh using the same r makes the dense samples as consistent as possible on the two surfaces. This joint sampling process yields consistent samples, especially if source and target regions correspond to similar shape parts. We denote the dense sample sets on regions due to S_l and T_l by $\hat{S}_l$ and $\hat{T}_l$, respectively.

5.3. Dense region matching

We match $\hat{S}_l$ and $\hat{T}_l$ by using a fast minimum-weight perfect matching algorithm [Kol09], and denote the resulting dense map by $\hat{\S}_l$. To feed the algorithm, we build a cost matrix $\mathbf{C}$ where each entry c_{pq} is the isometric distortion of matching a source sample $\hat{s}_p \in \hat{S}_l$ to a target sample $\hat{t}_q \in \hat{T}_l$. We compute each c_{pq} based on the three correspondences available in the qualified safe map $\S_l^{(3)}$ by setting $c_{pq} = d_{\text{iso}}(\hat{s}_p, \hat{t}_q, \S_l^{(3)})$ via (2), which is expected to map $\hat{S}_l$ to $\hat{T}_l$ with low distortion if $\S_l^{(3)}$ is correct. Since the cardinalities of the disjoint sets must match for a perfect matching, if $|\hat{S}_l| \neq |\hat{T}_l|$, we introduce virtual vertices with connector

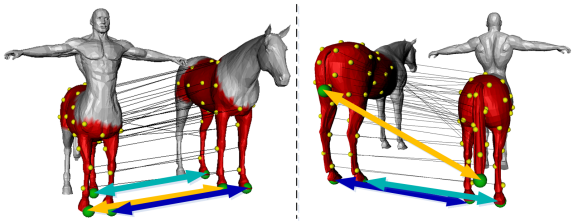

Figure 5: *Confidence vote assignment to the matches between the extremity samples (colored bold lines) for two different safe maps. Confidence votes are computed by traversing a pair (s_i,t_j) over the dense matching displayed with thin black lines.*

edges of ∞ weights. We also enforce the three correspondences available in the map $\S_l^{(3)}$ to be preserved in the resulting dense map by setting the corresponding entries of the cost matrix to $-\infty$. Hence we guarantee that $\S_l^{(3)} \subset \hat{\S}_l$.

5.4. Vote matrix

The dense region matching process described previously is repeated for each qualified safe map $\S_l^{(3)}$ generated from $G_m^{(3)}$, and each such matching process produces a confidence vote $\gamma_l(s_i,t_j)$ for each pair $(s_i,t_j) \in \S_l^{(3)}$. This confidence vote is computed based on the individual isometric distortion that the dense matching yields:

$$\gamma_l(s_i,t_j) = \exp(-d_{\text{iso}}(s_i,t_j,\hat{\S}_l)) \tag{4}$$

which produces a value in $[0,1]$.

The confidence votes resulting from all dense mappings are then accumulated into the vote matrix Γ, where each entry γ_{ij} eventually represents the confidence of matching a source extremity $s_i \in S$ to a target extremity $t_j \in T$. More specifically, each entry γ_{ij} is given by the average of all confidence votes that the pair (s_i,t_j) gets. We note that, to improve robustness, we discard a qualified safe map from the voting process if the target region of interest is significantly larger or smaller than the source region after scale normalization since this definitely implies a bad configuration, e.g., three source samples from finger tips of a hand vs. a target triplet consisting of two hands and a head on a pair of human shapes. In our experiments, we have discarded the cases where the target region is twice larger or smaller than the source. We have manually selected this setting so as to keep good configurations while eliminating those which are definitely bad.

6. Combining

We use the vote matrix Γ to find an optimal mapping, $\S^* : S \to T$, from the set of source extremities to the set of target extremities. We first convert the vote matrix into a cost matrix and then apply the minimum-weight perfect matching algorithm in [Kol09], that gives us an optimal one-to-one mapping which respects confidence values globally. The cost matrix $\mathbf{C}^*$ is formed by replacing the high confidence entries in Γ with $c^*_{ij} = 1 - \gamma_{ij}$ and others with ∞. Note that virtual vertices are introduced if $|S| \neq |T|$, as in Section 5.3. High-confidence entries are determined automatically using a procedure that is similar to the jump detection algorithm described in Section 4.2. Given a sample s_i, we sort all confidences in row Γ_i to infer the average difference ζ_i between the consecutive sorted confidences. We then mark the entries appearing before the first significant jump, which we set to $1.5\zeta_i$, as high-confidence entries. Taking into account only high confidence entries improves the robustness of the matching algorithm. We note that the choice $1.5\zeta_i$ is set manually by experimenting, which is a quite stable setting for thresholding confidence values.

The above perfect matching algorithm produces a one-to-one mapping that associates every source extremity with one target extremity sample. This is a desirable solution in the case of complete shape correspondence as well as for the problem of part matching. However, when the shapes are partially isometric both with parts that are not similar, some of the matches in the resulting map will clearly be outliers which distract the optimization process itself. Also, when the structural dissimilarity between the shapes is large, there is the danger of occupying a nice spot on the target shape with an irrelevant match which originates from a source sample whose counterpart does not actually exist on the target. To address this problem, we iterate the perfect matching algorithm each time removing one of the outliers. Since an outlier match is expected to have small confidence, at each iteration, we remove the least-confident match by setting the corresponding entry in $\mathbf{C}^*$ to ∞ and solving the new $\mathbf{C}^*$ again and repeating these removals until convergence, i.e., until there is no jump in the confidences of the matches in the resulting one-to-one map, i.e., all differences between consecutive sorted confidences are less than 3ζ, where ζ is the average of all these differences (we choose the setting 3ζ manually by experimenting). Hence the final map that our algorithm produces is always one-to-one, but does not necessarily associate every extremity sample on the source (or target) shape with an extremity on the other.

7. Extension to Dense Map

The optimal coarse correspondence $\S^*$ that our RAVAC algorithm produces between sparse shape extremities can be extended to a dense map. For each mapping with cardinality three, which is a subset of $\S^*$, we densely resample and match the corresponding regions of interest. This process is repeated for all $\S_l^{(3)} \subset \S^*$, and then the resulting dense matchings are blended into one dense map, that we denote by $\hat{\S}^*$.

The process of resampling and matching the regions is the same as described in Section 5 except that this time the resampling algorithm takes into account the other overlapping regions of interest while populating its samples. The regions of interest are enforced to include the same samples in the

parts where they overlap. Hence while resampling a region (Section 5.2), the dense sample set is initialized to include all the dense samples that have been so far included by some other regions of interest. This enables to accumulate a set of candidate matches on the target, $F(\hat{s}_i)$, for a given dense source sample $\hat{s}_i$ as regions of interest are matched. Let $\hat{\mathbf{t}}_j$ be the coordinate vector for the target dense sample $\hat{t}_j$. The blended coordinate

$$\mathbf{b}_i = \frac{1}{|F(\hat{s}_i)|} \sum_{\hat{t}_j \in F(\hat{s}_i)} \hat{\mathbf{t}}_j \tag{5}$$

then approximates the geodesic centroid of the candidate matches for $\hat{s}_i$ and provides $(\hat{s}_i, \hat{t}_k)$ as the blended dense match, where $\hat{t}_k$ is the target vertex closest to $\mathbf{b}_i$ in L_2 sense. The main computational load of this dense extension comes from the minimum-weight perfect matching phase, which is negligible when the number of dense samples is less than 500.

8. Computational Complexity

Sampling N initial extremities on the input mesh with V vertices takes $O(V \log V)$ time. The ranking module demands $O(N^4)$ operations as each of N samples is tested with all $O(N^3)$ triplets to traverse a map of constant size. The voting procedure, for each qualified triplet ($O(N^3)$), generates a potentially safe map, defines regions of interest around the map ($O(V)$) and distributes $\sim$100 dense samples on them ($O(V \log V)$), which are then matched for confidence computations. The voting complexity is hence $O(N^3 V \log V)$. The final combining phase performs minimum-weight perfect matching of $O(N^2 \log N)$ work about at most 10 times until convergence. The dense map extension comes without any additional complexity as the blended coordinates computed in $O(N^3 V \log V)$ time provide the closest mesh vertices in $O(NV)$ time. The overall worst case complexity is therefore $O(N^3 V \log V)$ assuming $N \ll V$.

Compared to $O(V^2 \log V + N^4 \log N)$ complexity of the MV method, our method is considered to be fast since it uses a much smaller N, e.g., 10 vs. 250. The BIM method, on the other hand, has the same algorithmic complexity as our method, yet not addressing the partial matching problem.

9. Experimental Results

We test the performance of our method on several shape benchmarks for partial, complete, sparse, and dense correspondence problems in the presence of isometric (or nearly isometric) deformations and scale differences. We mainly compare our method with the Möbius Voting (MV) method of [LF09] since our focus is on the partial correspondence problem. We also conduct experiments for comparison with the Blended Intrinsic Maps (BIM) method of [KLF11] in the case of complete dense correspondence.

The first dataset that we use is a subset of the Non-rigid World benchmark [BBK06], which consists of uniformly-sampled meshes representing articulated motions of 17 horses, 6 centaurs, 6 seahorse, 21 gorillas, 4 males, and 4 females, each with $\sim$3.4K vertices and arbitrary connectivity. We have also created 4 partial horse models by manually cropping the original complete models. The second dataset is a part of the SHREC'11 benchmark [BBB*11]. A high-resolution mesh of a null reference male model in T-pose (SHREC-null), its 5 different poses that have undergone isometric deformations (SHREC-iso), one isometric pose in 5 different scales (SHREC-sca), and 5 cropped models (SHREC-part) are represented with $\sim$50K uniformly spaced and arbitrarily connected vertices. The third dataset is the SCAPE benchmark [ASK*05], which is reconstructed from a real scene, representing the real motion of a human actor in 71 meshes each with $\sim$ 12.5K vertices. Finally we use high-resolution TOSCA shape benchmark [BBK08] with full ground-truth correspondence information for our dense matching experiments in comparison to BIM and MV.

Beside visual evaluations, we assess the performance using the distortion measure D_{grd} which quantifies the deviation of a given correspondence $\S$ from the ground-truth correspondence:

$$D_{\text{grd}}(\S) = \frac{1}{|\S|} \sum_{(s_i,t_j)\in\S} g(f(s_i), t_j), \tag{6}$$

where $f(s_i)$ stands for the ground-truth correspondence of s_i on the target shape, and $g(.,.)$ is the geodesic distance function. The maximum geodesic distance on the target model is normalized to 1.0 to simplify the interpretation of this measure. We note that dense ground-truth correspondences are available with SCAPE and TOSCA datasets whereas for SHREC'11 and Non-rigid World benchmarks we obtain the coarse ground-truth correspondences between shape extremities by hand.

9.1. Sparse Extremity Matching

We consider three possible sparse matching scenarios: complete matching, partial matching and part matching. Figures 6-10 display various examples from our sparse correspondence results. In these figures, we give the most confident 6 extremity matches in red, green, blue, black, cyan, and magenta colors, respectively, and for the subsequent matches, if exist, we use dashed black lines with spherical endpoints scaled with a radius proportional to the confidence of the correspondence pair. Unmatched samples, if exist, are represented by small red spheres.

In Table 1, we provide the quantitative performance results obtained on various benchmarks, in comparison to the MV method. For selection of the shape pairs in each test suit given in this table, we employ a randomized protocol which is similar to the one used in [KLF11]. For the complete matching tests between the Non-rigid World horse models, each complete model is mapped to a random complete model, whereas the part matching performance is evaluated by finding maps between each cropped model and 4 random complete models, hence a total of 33 pairs (see

	Extremities (MV, Our method)	5-matches (MV-top5, Our counterparts)
Dataset	$(D_{\text{grd}}, D_{\text{grd}})$	$(D_{\text{grd}}, D_{\text{grd}})$
Horse↔Horse	(.189, .028)	(.014, .066)
Horse↔Horse-part	(.281, .045)	(.039, .089)
Centaur↔Horse	(.348, .046)	(.025, .133)
Seahorse↔Horse	(n/a, .071)	(n/a, n/a)
Centaur↔Human	(n/a, .078)	(n/a, n/a)
SHREC-iso↔SHREC-iso	(.053, .003)	(.002, .044)
SHREC-part↔SHREC-iso	(n/a, .049)	(n/a, n/a)
SHREC-part↔SHREC-sca	(n/a, .051)	(n/a, n/a)
Gorilla↔SHREC-null	(n/a, .065)	(n/a, n/a)
SCAPE↔SCAPE	(.182, .004)	(.007, .045)

Table 1: *Quantitative evaluation of our method in comparison with Möbius Voting (MV) [LF09].*

also Fig. 6). The remaining tests conducted on the Non-rigid World benchmark seek partial correspondences in the presence of uncommon parts, that complicates the problem further. In these experiments, we pick 6 centaurs and match each of them with 4 random models from horse, male, and female classes, hence a total of 72 pairs. Similarly, each seahorse is matched to 4 random horse models for another 24 pairs (see also Fig. 7). The SHREC'11 complete matching tests are performed by mapping each isometry class model to a random model from the same class. As far as the partial correspondence is concerned, each model in the partial class is matched with 3 random models from the isometry and scaling classes, hence a total of 35 pairs (see also Fig. 8). Finally, the quantitative performance values for SCAPE dataset are computed over 10 randomly selected shape pairs.

For comparison tests, we have run the publicly available code of MV with its default settings of 100 samples and 1M votes. In Table 1, we evaluate the performance of MV based on the samples that are closest to the extremity samples used by our algorithm. We also compare the correspondence formed by the top (most confident) 5 MV matches with our 5 corresponding matches. For the former case concerning extremity matches, our method significantly outperforms MV whereas, for the latter 5 matches that tend to be on non-extremities such as shape centers, we are almost on a par with (only slightly worse than) it. We note that our initial sparse correspondence between extremity samples needs to be extended to a denser one, as described in Section 7, in order to find our closest counterparts to the top 5 MV matches. The missing entries in the table for the rows including SHREC-part meshes with holes are due to sphere topology restriction of MV. The entry for Centaur↔Human pair is also missing since the similarity between the shapes is required to be more than 40% in the case of MV [LF09]. Note also that gorilla and seahorse meshes crash the public MV code due to its sphere topology requirement.

We observe in Table 1 that the performance of MV on

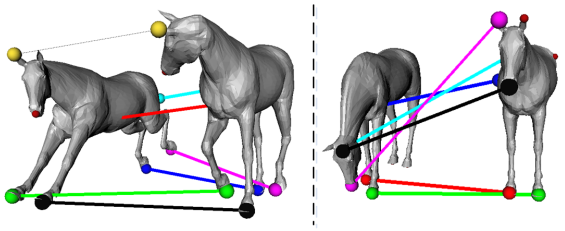

Figure 6: *Complete shape matching between the extremities of two horse models from Non-rigid World (left). Part matching between a horse and a cropped model (right).*

SCAPE models, which contain much more non-delaunay triangles than SHREC meshes, is inferior to its performance on SHREC although the object types and isometric deformations applied are quite similar for these two datasets. This decrease in performance is not observed in the case of our method which is insensitive to peculiarities of a given particular triangulation.

Several visual examples for comparison with MV are demonstrated in Fig. 9. The top 5 MV matches are highlighted by large spheres whereas their extremity matches that are closest to ours are indicated by large spheres with connecting lines. All other small spheres of matching colors represent the remaining correspondences. A similar visualization is performed for our results as well except that only 5 of our dense matches, which correspond to the top 5 MV matches, are shown. We observe that whenever the dissimilarity between shapes increases, MV shows instabilities especially at the extremity matches as the mutual closest point matches in their embedding domain starts to confuse on these regions of small area. Our results, on the other hand, rely on the dense matchings obtained in the neighborhoods of the extremities, which are hence less likely to get negatively affected by irrelevant data.

As we observe in Fig. 7, our algorithm mostly rules out the samples representing the uncommon parts without causing any confusions on the matches concerning samples of interest from the common parts. However we note that some of the correct matches, such as those between the heads of Seahorse↔Horse, may erroneously be removed by the iterative perfect matching process in the combining phase (Section 6) due to the consistent setting of the jump threshold value over all datasets, that is 3ζ.

We also experiment on a low-resolution gorilla and a high-resolution male from two different benchmarks to demonstrate the endurance of our algorithm not only to the difference and size of the triangulations but also to complete matching of shapes that exhibit local similarities but large deviations from global isometry (Fig. 10). With 21 pairs obtained by matching the null shape to all gorillas, we obtain successful results (see Table 1). Since we isolate each potentially compatible triplet pair from all other samples during

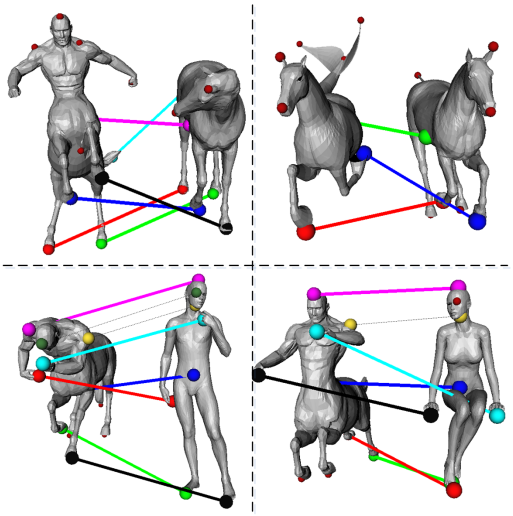

Figure 7: *Examples of partial matching on shape pairs from four shape classes of Non-rigid World (Horse, Centaur, Seahorse, and Human).*

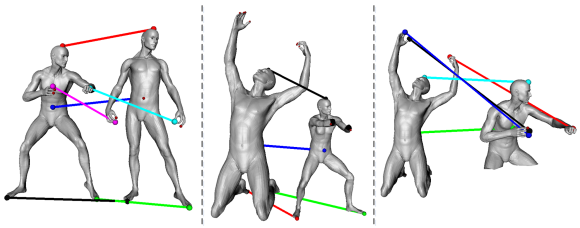

Figure 8: *Example matchings on SHREC'11 between two complete shapes from isometry class (left). Partial model mapped to scaling model (middle) and to isometry model (right). Note the arbitrary number of samples on fingers.*

the voting process, the male hands can be matched to the elongated gorilla hands successfully.

9.2. Dense matching

We evaluate our dense matching extension described in Section 7 in comparison to MV and BIM. For these tests, we use 8 Horse pairs from TOSCA and 12 pairs from SCAPE, all randomly selected. In this case, we can thoroughly evaluate D_{grd} over all matches, rather than on just 5 matches, thanks to the ground-truth dense correspondence information available.

Fig. 11 visualizes an example from our dense map of size ∼250, obtained on SCAPE dataset along with the corresponding MV map of the same size. The figure demonstrates our much smoother correspondence flow as compared to MV, where the yellow bold lines represent the worst individual matches, exemplifying the poor performance of MV around shape extremities. The quantitative evaluation also favors our method as given in Table 2. We also note that

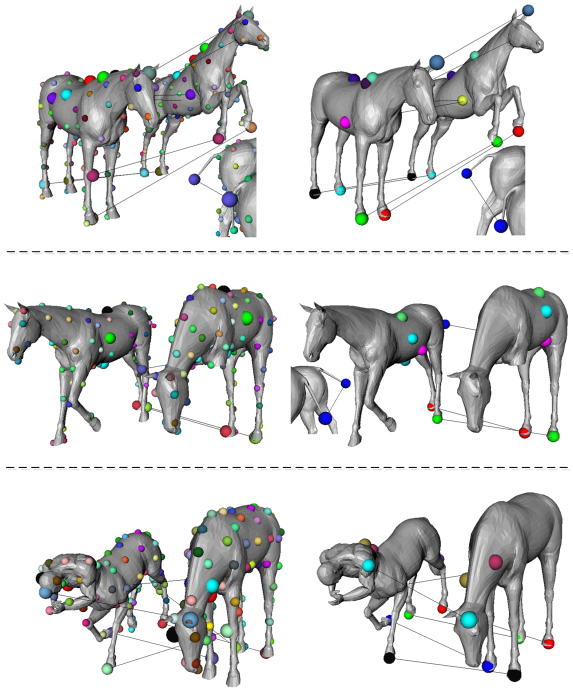

Figure 9: *Möbius Voting (left) vs. our method (right) on different shape classes: Horse-Horse, Horse-Horse part, and Centaur-Horse, for complete, part, and partial matching.*

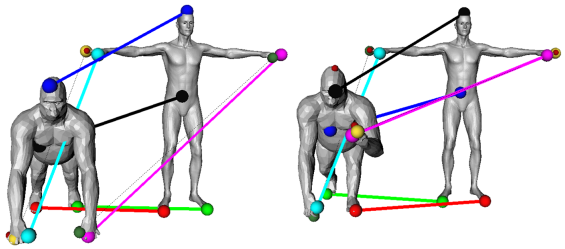

Figure 10: *Two examples of complete shape matching between male (SHREC'11) and gorilla (Non-Rigid World) meshes that exhibit local similarities but large deviations from global isometry.*

the worst distortions for the dense correspondences obtained on the 12 SCAPE pairs are .952 and .257 in our favor, and that the performance difference is higher on SCAPE meshes which contain more non-delaunay triangles than TOSCA horse models. Although our method does not in general yield very large errors as in MV, it is possible to have locally incorrect matches due to inconsistent sampling, such as nose to ear matching demonstrated in Fig. 12-top.

As for comparison with BIM, we have used the publicly available BIM code which produces a full dense map between input meshes. Hence for a fair comparison we interpolate our dense correspondence to a full map between all

	Extremities (MV, Our method)	Dense map (∼250) (MV, Our method)
Dataset	$(D_{\text{grd}}, D_{\text{grd}})$	$(D_{\text{grd}}, D_{\text{grd}})$
Horse↔Horse (TOSCA)	(.111, .024)	(.060, .031)
SCAPE↔SCAPE	(.223, .017)	(.203, .043)

Table 2: *Quantitative evaluation of our method in comparison with Möbius Voting method of [LF09].*

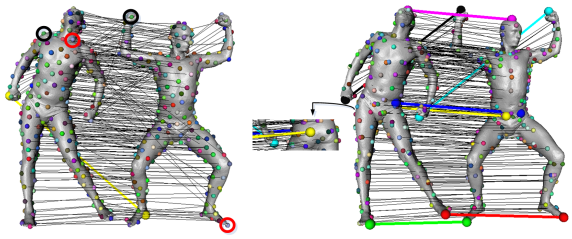

Figure 11: *Dense complete maps computed on a SCAPE pair by Möbius Voting (left) and our method (right). Some MV matches that correspond to our extremity matches are marked with circles. Yellow lines show the worst matches.*

vertices using the same procedure described in [KLF11] (see Fig. 12). In Table 3, we provide the extremity and full dense matching performances of BIM in comparison to ours. For extremity matching comparison, we follow the same strategy that we have used for MV comparisons given in Section 9.1. We observe that BIM is slightly better than our method in extremity matching, mainly because we enforce three extremity correspondences in the generating partial maps to be preserved in the resulting dense maps to be blended. The BIM method, on the other hand, blends unrestricted match candidates for a given extreme sample, that renders it more flexible as exemplified via nose matches in Fig. 12-top. Full dense maps of BIM are again slightly better than our interpolated counterparts. This performance difference in favor of BIM during these complete correspondence tests is actually as expected since our main concern is the more challenging partial correspondence problem that cannot be handled by the BIM method as we demonstrate next.

We emphasize that BIM has been designed specifically for complete shape matching; yet one could envisage using it for partial matching since it blends the generated complete maps by confidence weighting. Hence as a final set of experiments, we have tested BIM for its possible use in partial and part matching, as shown in Fig. 13. We have used the same publicly available BIM code with the same settings as we have used in the complete correspondence experiments. We observe that BIM is unstable in these scenarios and can generate incorrect pairings. This is mainly due to uncommon or dissimilar parts that constrain the distortion estimate and hence the weights of the blending process. Note also in Fig. 13 that the output is severely dependent on the choice that assigns one shape as source and the other as target. For the part matching scenario for instance, we do not require

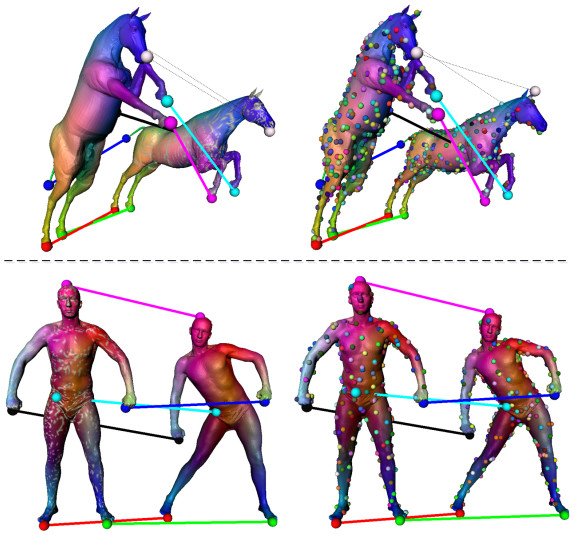

Figure 12: *Blended Intrinsic Maps (left) vs. our method (right) for complete correspondence on two different shape classes, (top row) TOSCA-Horse and (bottom row) SCAPE. The color of each source vertex is transferred to the corresponding target vertex where unmatched vertices are painted in grey. Our dense map used in interpolation is shown as spheres of matching colors.*

	Extremities (BIM, Our method)	Full dense map (BIM, Our method)
Dataset	$(D_{\text{grd}}, D_{\text{grd}})$	$(D_{\text{grd}}, D_{\text{grd}})$
Horse↔Horse (TOSCA)	(.007, .024)	(.019, .037)
SCAPE↔SCAPE	(.012, .017)	(.042, .051)

Table 3: *Quantitative evaluation of our method in comparison with Blended Intrinsic Maps (BIM) [KLF11].*

to know the small or cropped mesh in advance since our method first establishes one-to-one maps between subsets of extremities, each of which then points to regions of interest to be densely matched in many-to-one fashion. The BIM method however directly seeks a many-to-one mapping from source mesh to target mesh, hence requiring a priori knowledge of the smaller shape to be used as the source before performing any partial matching if that is the intention.

9.3. Timing

The execution times of our shape correspondence algorithm (including dense matching) on a 2.53GHz PC are about 150, 175, 145, 91, 557, 1216, and 22 seconds for Horse↔Horse, Centaur↔Horse, Seahorse↔Horse, Centaur↔Human, SHREC-iso↔SHREC-iso, Gorilla↔Human, and SCAPE↔SCAPE, respectively. The relatively high execution time on Gorilla↔Human is mainly due to 15 samples in matching as opposed to the typical 10 samples for the others. The fastest runs are on SCAPE↔SCAPE pairs, dealing with only ∼6 samples. The percentage of the

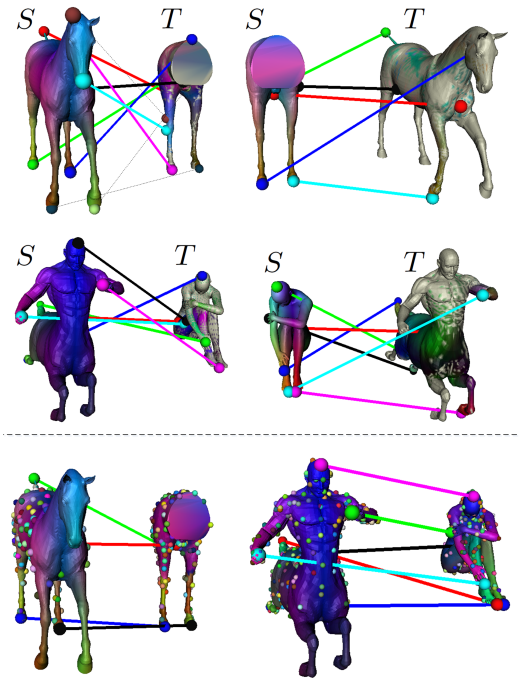

Figure 13: *Blended Intrinsic Maps with different source (S) choices (top two rows) vs. our method (bottom row), for partial matching on two different shape pairs from TOSCA (Horse↔Horse-part and Centaur↔Human). Same display format as Fig. 12.*

execution time devoted to each specific step of the algorithm, respectively for sampling, ranking, voting, combining and dense matching, is 11.1%, 0.9%, 85.4%, 0.4% and 2.2% on a SCAPE pair with 6 samples and ∼12.5K vertices; 0.4%, 9.6%, 89.1%, 0.8% and 0.1% on a Centaur-Horse pair with 11 vs. 9 samples and ∼3.4K vertices. We see that the execution times are dominated by the voting module which creates and samples regions of interests whereas the fast ranking phase just demands shortest path distances between few number of extremity samples.

9.4. Validation of the RAVAC algorithm

We now show by experiments the benefit of each additional step of our correspondence algorithm in terms of quantitative improvement. Specifically, we compare our original RAVAC algorithm with six modified versions as described below:

- Version 1 excludes both the voting and combining steps from the original algorithm (Version 0) by directly blending the partial dense maps $\{\hat{\delta}_{jl}\}$ as described in Section 7. Recall also from Section 5.3 that $\{\hat{\delta}_{jl}\}$ are originally designed to accumulate confidence values for the coarse extremity matches in the vote matrix (see Eq. 4).
- Version 2 excludes from the original algorithm the pruning step involved in ranking; it uses all possible one-to-one maps $\mathcal{S}^{(k)}$ for traversal in Eq. 3. Thus this version keeps ranking but without pruning.

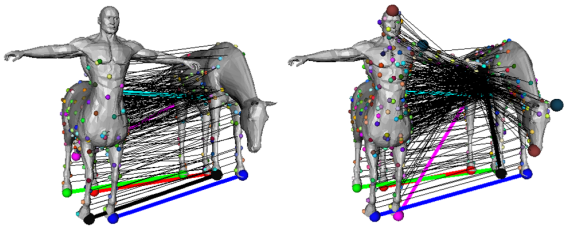

Figure 14: *Comparison of our original algorithm (left) to Version 1 (right) for Centaur-Horse partial matching.*

- Version 3 excludes the pruning step from Version 1 in a similar way as in Version 2.
- Version 4 excludes the ranking step altogether from the original algorithm. Hence all possible partial maps between triplets of extremities vote for matching pairs.

We perform the tests on SCAPE as well as the Horse and the Centaur classes from TOSCA, hence for the complete shape correspondence scenario, for which dense ground-truth correspondence information is available. The resulting ground-truth distortions (computed using the same randomized protocol as in Section 9.1 to select pairs from each class) are shown in Table 4 where we observe that the best performance is obtained using the original RAVAC algorithm.

The superior performance of RAVAC over direct blending versions (Versions 1 and 3) is mainly due to two reasons. The first one is the iterative outlier removal process involved in the combining step, that keeps only the reliable extremity matches to be used in blending. The second reason is the voting step that eliminates some of the (potentially safe) partial maps which may be incorrect or symmetrically flipped. Also note that the results given in Table 4 are obtained for the complete correspondence scenario. If the shapes to be matched contain uncommon surface parts, then the relative performance of the direct blending approaches becomes even worse since the outlier matches from uncommon parts cause much more error in overall extremity matching as well as in blending. We demonstrate this visually in Fig. 14 on an example where we compare the performance of Version 1 to our original algorithm in partial matching. We observe that, in the case of Version 1, outlier matches distract the whole correspondence process, resulting in an unsmooth and erroneous dense map.

Another important observation (Versions 2 and 3) is that the pruning step involved in the ranking phase not only reduces the computational load, but also increases the performance by pre-filtering a significant portion of the partial maps based on a simple geodesic consistency constraint. When the ranking step is omitted totally (Version 4), we see that the performance drop becomes very severe. This is as expected since the number of incorrect partial maps in consideration becomes significantly high in this case and so is the resulting distortion.

Dense map of size ∼250					
	v.0	v.1	v.2	v.3	v.4
Dataset	D_{grd}	D_{grd}	D_{grd}	D_{grd}	D_{grd}
Horse↔Horse	.042	.049	.051	.054	.203
Centaur↔Centaur	.059	.063	.070	.075	.227
SCAPE↔SCAPE	.053	.056	.066	.071	.224

Table 4: *Quantitative evaluation of our method in comparison with its modified versions (v.0 represents the original RAVAC algorithm).*

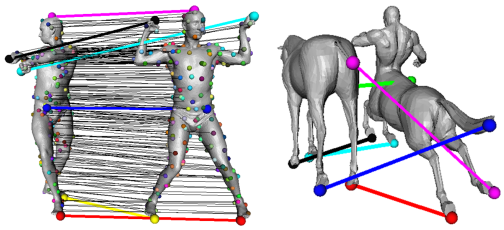

Figure 15: *A symmetrically flipped dense map (left) and an example to failure in detecting matches (right).*

9.5. Limitations

The most obvious limitation of the RAVAC algorithm is the approximate isometry requirement. Our method can be used to match shapes which contain surface parts that are approximately isometric, but fails to handle severe non-isometries.

Another limitation is that our method can confuse small-scale features that are close to each other, due to distraction of spurious samples as well as uncommon parts and local non-isometries, such as the nose-to-ear matching in Figures 6 and 12-top. In Fig. 15-right, we display a more severe failure case that occurs mainly due to the local non-isometry on the tails and imprecise setting of the jump threshold in the outlier removal process. As the jump threshold value, we currently use the same setting (3ζ) for all datasets, that may sometimes lead to incorrect removal of some matches such as those between the heads of Seahorse↔Horse in Fig. 7 as well as failures in detecting mismatches as in Fig. 15-right. Note that the mismatch in Fig. 15 is displayed in magenta color, meaning that it is the least confident match of the resulting map, hence it would be the first to be removed if the jump threshold were fine-tuned over the specific dataset. The blue leg-to-tail matching would then probably be corrected at the next iteration of the outlier removal process.

The use of geodesic distance metric for distortion computations can also be viewed as a limitation. Although our method generally performs well for shapes with holes such as SHREC-part meshes employed in our experiments with no restriction on shape topology, it may not be possible to compute geodesics reliably in the case of severe topological noise, and the algorithm may fail to generate accurate matchings.

The last limitation is due to the classical symmetrical flip problem which is actually inherent to all purely isometric correspondence techniques. We may hence occasionally end up with flipped results (see Fig. 15-left) for the cases where the dense analysis with ∼100 samples (see Section 5.2) remains insufficient to resolve intrinsic symmetries.

10. Conclusion

The basic assumption in our RAVAC algorithm is that two shapes can be matched based on their extremities. As long as this assumption holds, which is the case for partially isometric shapes, and these extremities can reliably be extracted, our algorithm produces correct matchings, which can then be extended to dense correspondence. This assumption can however also be seen as a restraining factor for the generality of our method to handle non-isometries as well as a possible source of inaccuracies since it makes our method sensitive to the performance of the extremity extraction process. Nevertheless, the experiments that we have conducted on various datasets show that our method performs reasonably well in the case of approximate isometries, and even for shapes with holes, such as SHREC-part meshes, with no restriction on shape topology. Some spurious extremities may appear in such cases but they are mostly handled thanks to our reliable voting approach and outlier elimination procedure. Provided that isometrically similar parts are represented by sufficient number of extremities (which is 3 at least), our method can match shapes exhibiting large deviations from global isometry, such as the Gorilla-Human pair from our experiments, or partially isometric shapes with quite small similarity overlap, such as the Centaur-Human pair.

The experiments that we have conducted show that our method outperforms the MV method of [LF09], the best performant algorithm available in the literature for partial shape correspondence. We once again iterate that the focus of this work is on partial correspondence, though the proposed algorithm can also generate complete correspondences. In the case of complete shape correspondence, the performance of our method is found to be worse than the BIM method of [KLF11], yet better than MV.

One possible way of further improving the results obtained by our method is to incorporate local shape descriptors, into the extremity sampling process in order to increase the precision and consistency of the samples, and/or into the cost (vote) matrix in order to increase the accuracy of the matching process. Another direction is to address the trade-off between the accuracy of the geodesic distortion metric currently in use and the topological noise robustness of the diffusion-based metrics as an alternative.

Acknowledgement

This work has been supported by TUBITAK under the project EEEAG-109E274.

References

[ACOT*10] AU O. K.-C., COHEN-OR D., TAI C.-L., FU H., ZHENG Y.: Electors voting for fast automatic shape correspondence. *Computer Graphics Forum (Proc. Eurographics) 29*, 2 (2010), 645–654. 2, 3

[ASK*05] ANGUELOV D., SRINIVASAN P., KOLLER D., THRUN S., RODGERS J., DAVIS J.: Scape: Shape completion and animation of people. *ACM Trans. Graph. 24*, 3 (2005), 408–416. 8

[BBB*11] BOYER E., BRONSTEIN A., BRONSTEIN M., BUSTOS B., ET AL.: SHREC 2011: Robust feature detection and description benchmark. *Eurographics Workshop on 3D Object Retrieval* (2011). 8

[BBK06] BRONSTEIN A. M., BRONSTEIN M. M., KIMMEL R.: Efficient computation of isometry-invariant distances between surfaces. *SIAM J. Scientific Computing 28*, 5 (2006). 2, 4, 8

[BBK08] BRONSTEIN A. M., BRONSTEIN M. M., KIMMEL R.: *Numerical Geometry of Non-Rigid Shapes*. Springer, 2008. 1, 8

[BK10] BRONSTEIN M. M., KOKKINOS I.: Scale-invariant heat kernel signatures for non-rigid shape recognition. *Proc. Computer Vision and Pattern Recognition (CVPR)* (2010). 2

[CH03] CARCASSONI M., HANCOCK E.: Spectral correspondence for point pattern matching. *Pattern Recognition 36* (2003), 193–204. 2

[DLL*10] DEY T. K., LI K., LUO C., RANJAN P., SAFA I., WANG Y.: Persistent heat signature for pose-oblivious matching of incomplete models. *Computer Graphics Forum (Proc. SGP) 29*, 5 (2010), 1545–1554. 2

[FS06] FUNKHOUSER T., SHILANE P.: Partial matching of 3d shapes with priority-driven search. *Computer Graphics Forum (Proc. SGP)* (2006). 2, 3

[GMGP05] GELFAND N., MITRA N., GUIBAS L., POTTMAN H.: Robust global registration. *Computer Graphics Forum (Proc. SGP)* (2005). 2

[HAWG08] HUANG Q., ADAMS B., WICKE M., GUIBAS L.: Non-rigid registration under isometric deformations. *Computer Graphics Forum (Proc. SGP)* (2008), 1149–1458. 2

[HSKK01] HILAGA M., SHINAGAWA Y., KOHMURA T., KUNII T.: Topology matching for fully automatic similarity estimation of 3d shapes. *Proc. SIGGRAPH* (2001). 3, 6

[JZ06] JAIN V., ZHANG H.: Robust 3d shape correspondence in the spectral domain. *IEEE Int. Conf. on Shape Modeling and Applications (SMI)* (2006), 118–129. 2

[KJS07] KRAEVOY V., JULIUS D., SHEFFER A.: Model composition from interchangeable components. *Proc. Pacific Graphics* (2007). 2

[KLCF10] KIM V. G., LIPMAN Y., CHEN X., FUNKHOUSER T.: Möbius transformations for global intrinsic symmetry analysis. *Computer Graphics Forum (Symposium on Geometry Processing) 29*, 5 (2010). 2

[KLF11] KIM V., LIPMAN Y., FUNKHOUSER T.: Blended intrinsic maps. *Proc. SIGGRAPH 30*, 4 (2011). 2, 8, 10, 11, 13

[Kol09] KOLMOGOROV V.: Blossom V: A new implementation of a minimum cost perfect matching algorithm. *Mathematical Programming Computation (MPC)* (2009), 43–67. 6, 7

[LAAD11] LIPMAN Y., AL-AIFARI R., DAUBECHIES I.: The continuous procrustes distance between two surfaces. *Communications in Pure and Applied Mathematics* (2011). 2

[LF09] LIPMAN Y., FUNKHOUSER T.: Möbius voting for surface correspondence. *ACM Trans. Graph. 28*, 3 (2009). 2, 3, 8, 9, 10, 13

[MHK*08] MATEUS D., HORAUD R., KNOSSOW D., CUZZOLIN F., BOYER E.: Articulated shape matching using laplacian eigenfunctions and unsupervised point registration. *Proc. Computer Vision and Pattern Recognition (CVPR)* (2008), 1–8. 2

[OBCS*12] OVSJANIKOV M., BEN-CHEN M., SOLOMON J., BUTSCHER A., GUIBAS L.: Functional maps: A flexible representation of maps between shapes. *Proc. SIGGRAPH* (2012). 3

[OMMG10] OVSJANIKOV M., MÉRIGOT Q., MÉMOLI F., GUIBAS L.: One point isometric matching with the heat kernel. *Computer Graphics Forum 29*, 5 (2010), 1555–1564. 2

[PBB11] POKRASS J., BRONSTEIN A. M., BRONSTEIN M. M.: A correspondence-less approach to matching of deformable shapes. *Proc. Scale Space and Variational Methods* (2011). 2

[PBB*13] POKRASS J., BRONSTEIN A. M., BRONSTEIN M. M., SPRECHMANN P., SAPIRO G.: Sparse modeling of intrinsic correspondences. *Computer Graphics Forum* (2013). 3

[ROA*13] RUSTAMOV R., OVSJANIKOV M., AZENCOT O., BEN-CHEN M., CHAZAL F., GUIBAS L.: Map-based exploration of intrinsic shape differences and variability. *Proc. SIGGRAPH* (2013). 3

[SY11] SAHILLIOĞLU Y., YEMEZ Y.: Coarse-to-fine combinatorial matching for dense isometric shape correspondence. *Computer Graphics Forum (Proc. SGP) 30*, 5 (2011), 1461–1470. 2, 4, 6

[SY12a] SAHILLIOĞLU Y., YEMEZ Y.: Minimum-distortion isometric shape correspondence using EM algorithm. *IEEE Trans. PAMI 34*, 11 (2012), 2203–2215. 2, 4

[SY12b] SAHILLIOĞLU Y., YEMEZ Y.: Scale normalization for isometric shape matching. *Computer Graphics Forum (Proc. Pacific Graphics) 31*, 7 (2012). 2, 3

[SY13] SAHILLIOĞLU Y., YEMEZ Y.: Coarse-to-fine isometric shape correspondence by tracking symmetric flips. *Computer Graphics Forum 32*, 1 (2013), 177–189. 4, 5

[TBW*09] TEVS A., BOKELOH M., WAND M., SCHILLING A., SEIDEL H.-P.: Isometric registration of ambiguous and partial data. *Proc. Computer Vision and Pattern Recognition (CVPR)* (2009). 2, 3

[vKZH13] VAN KAICK O., ZHANG H., HAMARNEH G.: Bilateral maps for partial matching. *Computer Graphics Forum, to appear* (2013). 2

[vKZHCO11] VAN KAICK O., ZHANG H., HAMARNEH G., COHEN-OR D.: A survey on shape correspondence. *Computer Graphics Forum 30*, 6 (2011), 1681–1707. 1

[WBBP11] WANG C., BRONSTEIN M. M., BRONSTEIN A. M., PARAGIOS N.: Discrete minimum distortion correspondence problems for non-rigid shape matching. *Proc. Scale Space and Variational Methods* (2011). 2

[ZBVH09] ZAHARESCU A., BOYER E., VARANASI K., HORAUD R.: Surface feature detection and description with applications to mesh matching. *Proc. Computer Vision and Pattern Recognition (CVPR)* (2009). 2

[ZSCO*08] ZHANG H., SHEFFER A., COHEN-OR D., ZHOU Q., VAN KAICK O., TAGLIASACCHIL A.: Deformation-driven shape correspondence. *Computer Graphics Forum* (2008), 1431–1439. 2, 3, 5

[ZWW*10] ZENG Y., WANG C., WANG Y., GU X., SAMARAS D., PARAGIOS N.: Dense non-rigid surface registration using high-order graph matching. *Proc. Computer Vision and Pattern Recognition (CVPR)* (2010). 2