SHREC’14 Track:
Shape Retrieval of Non-Rigid 3D Human Models


1Cardiff University, UK
2National University of Defense Technology, China
3Peking University, China
4Toyohashi University of Technology, Japan
5Technion - Israel Institute of Technology, Israel
6Northwestern Polytechnical University, China
7University of Verona, Italy
8National Institute of Standards and Technology, USA
9Concordia University, Canada
10Fraunhofer IDM@NTU, Singapore
11Texas State University, San Marcos, USA
12Penn State University, USA
13Beijing Technology and Business University, China
14University of Lugano, Switzerland
*Track organisers

Abstract
We have created a new benchmarking dataset for testing non-rigid 3D shape retrieval algorithms, one that is much more challenging than existing datasets. Our dataset features exclusively human models, in a variety of body shapes and poses. 3D models of humans are commonly used within computer graphics and vision, and so the ability to distinguish between body shapes is an important shape retrieval problem. In this track nine groups have submitted the results of a total of 22 different methods which have been tested on our new dataset.

1. Introduction
The ability to recognise a deformable object’s shape, regardless of the pose of the object, is an important requirement for modern shape retrieval methods. Many state-of-the-art methods achieve extremely high accuracy when evaluated on the most recent benchmark [LGB∗11]. It is therefore hard to distinguish between good methods, and there is little room to demonstrate improvement in approaches. There is thus a need for a more challenging benchmark for non-rigid 3D shape retrieval. Many novel approaches have been published since the previous benchmark, and therefore a new comparison of state-of-the-art methods is also beneficial.

We have created a new, more challenging, benchmarking dataset for testing non-rigid 3D shape retrieval algorithms.
2. Datasets

Our track uses two datasets, a Real dataset, obtained by scanning real human participants, and a Synthetic dataset, created using 3D modelling software. The latter may be useful for testing algorithms intended to retrieve synthetic data, with well sculpted local details, while the former may be more useful to test algorithms that are designed to work even in the presence of noisy coarsely captured data lacking in local detail.

2.1. Real Dataset

The Real dataset was built from point-clouds contained within the Civilian American and European Surface Anthropometry Resource (CAESAR) [caes]. This dataset comprises 400 meshes, representing 40 human subjects (20 male, 20 female) in 10 different poses. The point-cloud models were manually selected from CAESAR to be models with significant visual differences. We employed SCAPE (shape completion and animation of people) [ASK*05] to build articulated 3D meshes, by fitting a template mesh to each subject. Realistic deformed poses of each subject were built using a data-driven deformation technique [CLC*13]. We remeshed the models using freely available software [VC04, VCP08]. The resulting models have approximately 15,000 vertices.

2.2. Synthetic Dataset

We also used the DAZ Studio [DAZ13] 3D modelling/animation software to create a dataset of synthetic human models. The software includes a parametrized human model, where parameters control body shape. We used this to produce a dataset consisting of 15 different human models (5 male, 5 female, 5 child), each with its own unique body shape. We generated 20 different poses for each mode, resulting in a dataset of 300 models. The same poses were used for each body shape, and models are considered to belong to the same class if they share the same body shape. All models were remeshed using the same method as for the Real dataset. The resulting models have approximately 60,000 vertices. A selection of both real and synthetic models is shown in Figure 1.

3. Evaluation

We assessed two different retrieval tasks:

1. Returning a list of all models, ranked by shape similarity to a query model.
2. Returning a list of models that all share the same shape as the query model.

For both tasks, every model in the database was used as a separate query model. In the first task, for each query we asked the participants to order all other models in the dataset in terms of similarity to the query model. In the second task, for each query the participants were asked to submit a list of arbitrary length of all models which they classify as ‘the same shape’ as the query model. Both tasks were evaluated separately.

The evaluation procedure for Task 1 is identical to that used in several previous SHREC tracks [LGB*11]. We evaluated the results using various statistical measures: precision and recall, nearest neighbour (NN), first tier (1-T), second tier (2-T), e-measure (E-M), discounted cumulative gain (DCG), and precision and recall curves. Definitions of these measures are given in [SMKF04]. The results for Task 2 were evaluated using the F-Measure [BYRN11].

4. Methods

4.1. Simple shape measures, and Euclidean distance based canonical forms D. Pickup, X. Sun, P. L. Rosin and R. R. Martin

This section presents two techniques, simple shape measures based on surface area, and skeleton driven canonical forms.

4.1.1. Simple shape measures

Two simple shape measures were tested separately on the datasets. The first is the total surface area of the mesh, where parameters control body shape. We used this to produce a dataset consisting of 15 different human models (5 male, 5 female, 5 child), each with its own unique body shape. We generated 20 different poses for each mode, resulting in a dataset of 300 models. The same poses were used for each body shape, and models are considered to belong to the same class if they share the same body shape. All models were remeshed using the same method as for the Real dataset. The resulting models have approximately 60,000 vertices. A selection of both real and synthetic models is shown in Figure 1.
4.1.2. Skeleton driven canonical forms

A variant on the canonical forms presented by Elad and Kimmer [EK03] is used to normalise the pose of all the models in the dataset, and then the rigid view-based method by Lian et al. [LGSSX13] is used for retrieval. A canonical form is produced by extracting a curve skeleton from a mesh, using the method by Au et al. [ATC08]. The SMACOF Multidimensional Scaling method used by [EK03] is then applied to the skeleton, to put the skeleton into a canonical pose. The skeleton driven shape deformation method by Yan et al. [YMY08] is then used to deform the mesh to the new pose defined by the canonical skeleton. This produces a similar canonical form to [EK03], but with the local features better preserved. The models in the Synthetic dataset are simplified to approximately 15000 vertices, and any holes are filled, before computing the canonical form.

4.2. Hybrid shape descriptor and meta similarity

generation for non-rigid 3D model retrieval, B. Li, Y. Lu, A. Godil and H. Johan

A hybrid shape descriptor [LGJI13] has been proposed to integrate both geodesic distance-based global features and curvature-based local features. An adaptive algorithm based on Particle Swarm Optimization (PSO) is developed to adaptively fuse different features to generate a meta similarity between any two models. The approach can be generalized to similar approaches which integrate more or other features. It first extracts three component features of the hybrid shape descriptor: curvature-based local feature, geodesic distance-based global feature, and Multidimensional scaling (MDS) based ZFDR [LJon] global feature. Based on the extracted features, corresponding distance matrices are computed and they are fused into a meta distance matrix based on PSO. Finally, the distances are sorted to generate the retrieval lists.

Curvature-based local feature vector: \( V_c \), First, the Curvature Index feature of a vertex \( p \) is computed, which characterizes local geometry: \( CI = \frac{2}{\pi} \log (\frac{k_1 + k_2}{2}) \), where \( k_1 \) and \( k_2 \) are the two principal curvatures in the \( x \) and \( y \) directions respectively at \( p \). Then the Curvature Index deviation feature of the adjacent vertices of \( p \) is computed: \( \delta CI = \sqrt{\sum_{i=1}^{n} (CI_i - CI)^2} \), where \( CI_1, CI_2, \ldots, CI_n \) are the Curvature Index values of the adjacent vertices of \( p \) and \( CI \) is the mean Curvature Index of all the adjacent vertices. Next, to describe the local topological property, the Shape Index feature of \( p \) is computed: \( SI = \frac{2}{\pi} \arctan (\frac{k_1 - k_2}{k_1 + k_2}) \). After that, a combined local shape descriptor is formed by concatenating the above three local features: \( F = (CI, \delta CI, SI) \). Finally, based on the Bag-of-Words framework, the local feature vector \( V_C = (h_1, h_2, \ldots, h_N) \) is generated, where the number of cluster centres \( N_C \) is set to 50.

Geodesic distance-based global feature vector: \( V_G \), First, to avoid the high computational cost involved in the geodesic distance computation among many vertices, the models are simplified to 1000 vertices. Next, the geodesic distances among all the vertices of a simplified model are generated to form a geodesic distance matrix GDM. Finally, the GDM is decomposed based on Singular Value Decomposition and the first largest \( k \) eigenvalues are used as the global feature vector. In experiments, \( k \) is set to 50.

MDS-based ZFDR global feature vector: \( V_Z \), To leverage pose and deformation variations of non-rigid models, Multidimensional scaling (MDS) techniques are utilized to map the non-rigid models into a 3D canonical form. The previously computed geodesic distances among the 1000 vertices of each simplified 3D model are used as the input of MDS for the feature space transformation. Finally, the hybrid global shape descriptor ZFDR [LJon] is used to characterize the features of the transformed 3D model in the new feature space. There are four feature components in ZFDR: Zernike moments feature, Fourier descriptor feature, Depth information feature and Ray-based feature. This approach is named as MDS-ZFDR and Stress MDS is adopted in the experiments. It was also found that for 3D human retrieval using \( R \) feature only (that is MDS-R) can always achieve better results than other combinations such as ZF, DR or ZFDR. The reason should be related to the more salient feature of the geometry-related ‘thickness’ variations in the human models, such as fat versus slim bodies which are better characterized by the \( R \) feature, compared to other visual-related features like \( ZF \) and \( D \).

Retrieval algorithm: (1) Computation of Curvature-based local feature vector \( V_C \) based on the original models and local feature distance matrix \( M_C \) generation; (2) Computation of Geodesic distance-based global feature vector \( V_G \) and global feature distance matrix \( M_G \); (3) MDS-based ZFDR global feature vector \( V_Z \) and MDS-ZFDR global feature distance matrix \( M_Z \) computation; (4) PSO-based meta distance matrix generation and ranking. A meta distance matrix \( M = w_C M_C + w_G M_G + w_Z M_Z \) is generated, where \( w_C, w_G \) and \( w_Z \) fall in [0,1]. As a swarm intelligence optimization technique, PSO-based approach is robust and fast in solving problems that are non-linear and non-differentiable. It includes four steps: initialization, particles’ velocity and positions update, search evaluation and result verification. The number of particles \( N_P = 10 \); the maximum number of search iterations \( N_I = 10 \); and First Tier is selected as the fitness value for search evaluation. Please note that the PSO-based weight assignment preprocessing step is only performed once for each of the test sets.

The ‘Hybrid_R’ runs only use ‘MDS-R’ features, compared to the original ‘Hybrid’ approach presented in [LGJI13] which uses ‘MDS-ZFDR’. Besides comparing the component features including ‘Curvature’, ‘Geodesic’ distance and ‘MDS-ZFDR’ based features, the performance of ‘MDS-R’ is compared with ‘MDS-ZFDR’.
4.3. Histograms of Area Projection Transform, A. Giachetti and V. Garro

Human characters are recognised with the Histograms of Area Projection Transform (HAPT), general purpose shape descriptors proposed in [GL12]. The method is based on a spatial map (Multiscale Area Projection Transform) that encodes the likelihood of the points inside the shape of being centres of spherical symmetry. This map is obtained by computing for each radius of interest the value:

\[
\text{APT}(x, S, R, \sigma) = \text{Area}(T_R^{-1}(k_{R}(x) \subset T_R(S, \sigma)))
\]  

(1)

where \(S\) is the surface of interest, \(T_R(S, \sigma)\) is the parallel surface of \(S\) shifted along the normal vector (only in the inner direction) and \(k_{R}(x)\) is a sphere of radius \(\sigma\) centred in the generic point \(x\) where the map is computed. Values at different radii are normalized in order to have a scale-invariant behaviour, creating the Multiscale APT (MAPT):

\[
\text{MAPT}(x, y, z, S, R, \sigma(R)) = \alpha(R) \text{APT}(x, y, z, S, R, \sigma(R))
\]  

(2)

where \(\alpha(R) = 1/4\pi R^2\) and \(\sigma(R) = c \cdot R\ (0 < c < 1)\).

A discretized MAPT is easily computed, for selected values of \(R\), on a voxelized grid including the surface mesh, with the procedure described in [GL12]. The map is computed in a grid of voxels with side \(s\) on a set of corresponding sampled radius values \(R_1, \ldots, R_p\). In the paper it is also shown that histograms of MAPT computed inside the objects are very good global shape descriptors, showing very good performances on the SHREC 2011 Non-Rigid Watertight contest data [LGB11]. For that recognition task discrete MAPT maps were quantized in 12 bins and histograms computed at the different scales (radii) considered were concatenated creating a unique descriptor. Voxel side and sampled radii were chosen differently for each model and proportional to the cubic root of the object volume, in order to have the same descriptor for scaled versions of the same geometry. \(c\) was always taken equal to 0.5.

For the recognition of different human subjects, however, scale invariance is not wanted. For this reason a fixed voxel size and a fixed set of radii are used.

The values for these parameters have been chosen differently for the Real and the Synthetic datasets, using simple heuristics. The algorithm was tested using three different parameter configurations for each dataset (Real and Synthetic). The results were then compared, and the best configurations for each dataset were submitted to the track. The voxel size was taken similar to the size of the smaller details well defined in the meshes. For the Synthetic dataset, where fingers are visible and models are smaller, \(s = 4\ \text{mm}\) is used and 11 increasing radii have been computed starting from \(R_1 = 8\ \text{mm}\) and iteratively adding a fixed step of 4\ \text{mm} for the remaining values \([R_2, \ldots, R_1]\). For the Real dataset, where models are bigger and details are smoothed, \(s = 12\ \text{mm}\) is used applying 7 different radii starting from \(R_1 = 24\ \text{mm}\) with a constant radius increasing of 12\ \text{mm}.

The procedure for model comparison then simply consists in concatenating the histograms computed at the different scales and measuring distances between shapes by evaluating the Jeffrey divergence of the corresponding concatenated vectors.

In the tests this ‘general purpose’ shape comparison procedure is applied without specific adaptations to the task. A possible way to specialize it for human body recognition may consist in learning discriminative sets of radii with a feature selection procedure or in recognizing and comparing specific body regions.

The MAPT/histograms extraction (using the c++ implementation available at http://www.andreagiachetti.it) for the Real dataset takes around 46 min, with a mean of 7 sec. for each model; the computation for the Synthetic dataset is much longer dealing with more detailed meshes: 2 hours for the entire dataset, 25 sec. for each shape. A single query takes around 1.2 msec, using a Matlab implementation of the Jeffrey divergence distance.

4.4. R-BiHDM, J. Ye

The R-BiHDM [YYY13] method is a spectral method used for general non-rigid shape retrieval. Using modal analysis, the method projects Biharmonic distance [LRF10] map into a low-frequency representation which operates on the modal space spanned by the lowest eigenfunctions of shape Laplacian [RWP06, OBCT12], and then computes its spectrum as an isometric shape descriptor.

Let \(\psi_i, \psi_1, \ldots, \psi_m\) be the eigenfunctions of Laplacian \(\Delta\) corresponding to its smallest eigenvalues \(0 = \lambda_0 < \lambda_1 \leq \ldots \leq \lambda_m\). Let \(d(x, y)\) be the Biharmonic distance between two points on mesh, which is defined as:

\[
d(x, y)^2 = \sum_{i=1}^{m} \frac{1}{\lambda_i} (\psi_i(x) - \psi_i(y))^2.
\]  

(3)

The squared Biharmonic distance map \(D^2\) is a functional map defined by:

\[
D^2[f](x) = \int_{S \in S} d^2(x, f(y)) dy,
\]  

(4)

where \(S\) is the differential manifold of shape. The reduced matrix version of \(D^2\) is denoted by \(\text{A} = \{a_{ij}\}\), where \(a_{ij} = \left< \psi_i, \psi_j \right>_S = \int_S \psi_i(x)D^2\psi_j(x)dx\) for \(0 \leq i, j \leq m\). Note that \(\text{tr}(\text{A}) = 0\) and all eigenvalues of \(\text{A}\), denoted by \(\mu_0, \ldots, \mu_m\), are in a magnitude descending order, where \(\mu_0 > 0\) and \(\mu_i < 0\) for \(i > 0\). The shape descriptor is defined as a vector \([\mu_1, \ldots, \mu_m]\) (scale dependent) or \([\frac{\mu_1}{\mu_m}, \ldots, \frac{\mu_1}{\mu_m}], [\mu_1, \ldots, \mu_m]^T\) (scale independent). For this shape contest, we choose \(L = 30\) and \(m = 100\). Finally, a normalized Euclidean distance is used for nearest neighbour queries. The descriptor is insensitive to a number of perturbations, such as isometry, noise, and remeshing. It has superior discrimination capability regarding globally change of shape and is very efficient to com-

submitted to Eurographics Workshop on 3D Object Retrieval (2014)
It has been shown that scale independent descriptor (R-BiHDM) is more reliable for generic nonrigid shape tasks, while scale dependent descriptor (R-BiHDM-s) is more suitable for this human shape task.

4.5. HKS-TS and SIHKS-H, L. Lai, X. Lui and H. Li

The HKS-TS (heat kernel signature based on time serial) is an application of HKS [SOG09], which adds the statistics of dynamic HKS on a shape according to an appropriate time serial chosen using a subset of the Real data. The SIHKS-H (scale invariant heat kernel signature based on statistics histogram) is an application of SI-HKS [BK10]. The SI-HKS is calculated on the shape to form a histogram. Then the similarity between different shapes can be calculated according to the SIHKS-H. Different similarity will be found by using different methods. Finally, the ranking list can be produced according to the similarity. For Task 2 two methods are used, HKS-TS-HC and SIHKS-H-CC. They add a further processing step to the methods used for Task 1. HC means the hierarchical clustering algorithm. The hierarchical clustering algorithm is added for classification according to the similarity calculating in Task 1. Intuitively, the algorithm is improved to be fit for this task.

4.6. High-level Feature Learning for 3D Shapes, S. Bu, S. Chen, Z. Lui and J. Han

The proposed high-level feature learning method for 3D shapes is carried out in the following three stages.

1. Low-level feature extraction: three representative intrinsic features, scale-invariant heat kernel signature (SI-HKS) [BK10], shape diameter function (SDF) [GSCO07], and averaged geodesic distance (AGD) [HSKK01], are adopted as low-level descriptors.

2. Middle-level feature extraction: To tackle the spatial information missing in the low-level features, a middle-level position-independent Bag-of-Features (BoF) is first extracted from the above low-level 3D descriptors. In order to compensate the lack of structural relationship, the original BoF is further extended into a geodesics-aware BoF (GA-BoF), which considers the geodesic distance between each pair of BoF on the 3D surface.

3. High-level feature learning: Finally, a deep learning based approach is introduced to further learn high-level features from the GA-BoF, which is able to discover the intrinsic relationship among GA-BoF and provide high discriminative features for 3D shape retrieval.

4.6.1. Low-level 3D Shape Descriptors

In this research, the scale-invariant heat kernel signature, shape diameter function, and average geodesic distance are adopted as the low-level 3D shape descriptors which are used for generating middle-level features, since these three local descriptors are robust against non-rigid and complex shape deformations. The first six frequency components of the SI-HKS, SDF and AGD descriptors are concatenated to form a low-level shape descriptor as

$$F(x) = \{\text{SIHKS}(x_i) \mid i \geq 1, \text{SDF}(x_i), \text{AGD}(x_i)\},$$

where the dimension of the feature is $M = 8$.

4.6.2. Middle-level Features

In this step, Bag-of-Features (BoFs) are computed to represent the occurrence probability of geometric words, and Minkowski metric is adopted as feature weighting [CdAM12] for k-means to generate geometric words more precisely.

After the geometric words $\mathcal{C} = \{c_1, c_2, \ldots, c_K\}$ of size $K$ is obtained, the next step is to quantize the low-level descriptor space in order to obtain a compact representation. For each point $x \in X$ with the descriptor $F(x)$, feature distribution $\phi(x)$ is defined as

$$\phi_i(x) = c(x) \exp \left( -\frac{||F(x) - c_i||^2}{k_{BoF} \sigma_{min}^2} \right),$$

where the constant $c(x)$ is selected to satisfy $||\phi(x)||_1 = 1$.

The geodesics on the mesh are used to measure the spatial relationship between each pair of BoFs on vertices, and introduce the geodesics-aware Bag-of-Features (GA-BoF):

$$v(X) = \sum_{x_i \in X} \sum_{x_j \in X} \phi_i(x_i) \phi_j(x_j) \exp \left( -k_{gd} \frac{g_{dj}(x_i, x_j)}{\sigma_{gd}} \right),$$

where $N(x)$ is a normalization factor which makes features have a fixed maximum value of 1, $\sigma_{gd}$ is the maximal geodesic distance of any pair of vertices on the mesh, and $k_{gd}$ denotes the decay rate of distances, which is selected empirically. The resulting $v$ is a $K \times K$ matrix, which represents the frequency of geometric words $i$ and $j$ appearing within a specified geodesic distance. This expression provides occurrence probability of geometric words and relationship between them.

4.6.3. Feature Learning via Deep Learning

In order to further deeply mine the relationship of features from intra-class shapes and inter-class shapes in a large dataset, deep learning is introduced into our framework, which will result in high-level features with strong generalization. Due to the fact that deep belief networks (DBN) [HOT06] has shown good performance and is a probabilistic approach, DBN is adopted as the feature learning method to extract high-level features for the 3D shapes.

Stacking a number of the restricted Boltzmann machines (RBMs) and learning layer by layer from bottom to top gives rise to a DBN. It has been shown that the layer-by-layer greedy learning strategy [HOT06] is effective, and the greedy procedure achieves approximate maximum likelihood learning. In this method, the bottom layer RBM is
trained with the input data of GA-BoF, and the activation probabilities of hidden units are treated as the input data for training the upper-layer RBM, and so on.

In the shape retrieval task, unlabelled 3D shape data are used to train the DBN layer-by-layer. After obtaining the optimal parameters, the input GA-BoF’s are processed layer-by-layer till the final layer which are used as the high-level features. In the retrieval, $L_2$ distance of the features is used to measure the similarity of two shapes $X$ and $Y$ as

$$d_2(X, Y) = \|o(X) - o(Y)\|_2. \quad (8)$$

### 4.7. Bag-of-Features approach with Augmented Point Feature Histograms

The developed Augmented Point Feature Histograms (APFH) expands Point Feature Histograms (PFH) [RMBB08] by adding the statistics of their geometric features. PFH is known as a local feature vector for 3D point clouds. PFH constructs a histogram of geometric features extracted from neighbouring oriented points. Improving the discriminant power of PFH by adding the mean and covariance of its geometric features is investigated. Because APFH is a local feature vector as well as PFH, it is invariant to the global deformation and articulation of the 3D model.

The overview of how the method defines the proposed APFH is illustrated in Figure 2. With APFH, the first step is to randomly generate oriented points on the triangle surface of a 3D model using Osaka’s method [OFCD02]. To generate a random point $p$ on an arbitrary triangle surface composed of vertices $v_a, v_b,$ and $v_c$, the following formula is employed:

$$p = (1 - \sqrt{r_1})v_a + \sqrt{r_1}(1 - r_2)v_b + \sqrt{r_1}r_2v_c. \quad (9)$$

In the implementation, two random variables, $r_1$ and $r_2$ in the above equation, are computed using the Niederreiter pseudorandom number generator [BFN94]. The oriented point is generated by inheriting the normal vector of the surface as an orientation of the point.

Next a PFH for each oriented point is constructed. The PFH finds the $k$-neighbourhood for each oriented point, and calculates a four-dimensional geometric feature $f = [f_1, f_2, f_3, f_4]^T$ as proposed in [WHH03]. The four-dimensional geometric feature is defined as follows for every pair of points $p_a$ and $p_b$ in the $k$-neighbourhood, and for their normal vectors $n_a$ and $n_b$:

$$f_1 = \arctan(w \cdot n_b, u \cdot n_a),$$
$$f_2 = v \cdot n_b,$$
$$f_3 = u \cdot \frac{p_b - p_a}{d},$$
$$f_4 = d,$$

where $u = n_a, v = (p_b - p_a) \times u/||p_b - p_a \times u||, \quad w = u \times v,$ and $d = ||p_b - p_a||$. The PFH collects the four-dimensional geometric features in a 16-bin histogram $f_h$. The index of the histogram bin $h$ is defined by the following formula:

$$h = \sum_{i=1}^{4} s(t, f_i) \cdot 2^{i-1},$$

where $s(t, f)$ is a threshold function defined as 0 if $f < t$ and 1 otherwise. The threshold value of $f_1$, $f_2$, and $f_3$ are set to 0, and set the threshold value of $f_4$ to the average value of $f_4$ in the $k$-neighbourhood.

Furthermore, the mean and covariance of the four-dimensional geometric features is calculated. Let $f_m$ be the four-dimensional geometric feature of an oriented point in the $k$-neighbourhood. The mean feature $f_m$ and covariance feature $f_c$ in the $k$-neighbourhood are defined as follows:

$$f_m = \frac{1}{k} \sum_{i=1}^{k} f_i,$$
$$f_c = \text{Upper} \left( \frac{1}{k-1} \sum_{i=1}^{k} (f_i - f_m)(f_i - f_m)^T \right),$$

where $\text{Upper}(\cdot)$ concatenates the upper triangular part of the matrix. Our APFH $f_{APFH}$ is composed $f_m, f_c,$ and $f_c$.

Finally, APFH $f_{APFH}$ is normalized with the power and the $L_2$ normalization [PSM10].

To compare 3D models, the set of APFH features of a 3D model is integrated into a feature vector of the 3D model using the Bag-of-Features (BoF) approach [BBGO11, SZ03]. Moreover, the BoF is projected onto Jensen-Shannon kernel space using the homogeneous kernel map method [VZ12]. This approach is called BoF-APFH.

In addition, similarity between features is calculated using the manifold ranking method with the unnormalized graph Laplacian [ZBS11]. This approach is called MR-BoF-APFH.

The parameters of each algorithm are fixed empirically. For the APFH, the number of points is set to 20000, and the number of the neighbourhood to 55. For the BoF-APFH approach, a codebook of 1200 centroids is generated using $K$-means clustering, and used the SHREC’11 Non-rigid 3D Watertight dataset for training of the codebook.

### 4.8. BoF and SI-HKS, R. Litman, A. Bronstein, M. Bronstein and U. Castellani

All shapes were down-sampled to have 4,500 triangles. For each shape $S$ in the data-set, an SI-HKS [BK10] descriptor $x_i$ was calculated in every point $i \in S$. Unsupervised dictionary learning was done over randomly selected descriptors from all of the shapes using the SPAMS toolbox [MBPS09], with dictionary size of 32. The resulting 32 atom dictionary $D$ was, in essence, the bag-of-features of this method. Next, in every point descriptor $x_i$ was ‘replaced’ with a sparse code $z_i$ by solving pursuit problem.

submitted to Eurographics Workshop on 3D Object Retrieval (2014)
Spectral graph wavelet signature: The first stage consists of the computation of a dense spectral descriptor \( h(x) \) at each vertex of the triangle meshed shape \( X \). In general, any one of spectral descriptors with the eigenfunction-squared form reviewed in [LH13c] can be used in the human body retrieval contests for isometric invariant representation. In this work the recently proposed spectral graph wavelet signature (SGWS) is used as the local descriptor; it provides a general and flexible interpretation for the analysis and design of spectral descriptors \( s_k(t,x) = \sum_{i=0}^{m} g(t, \lambda_i) \phi_i(x) \). In a bid to capture the global and local geometry, a multi-resolution shape descriptor was obtained by setting \( g(t, \lambda_i) \) as a cubic spline wavelet generating kernel and considering the scaling function. The resolution level is set as 2.

Intrinsic spatial pyramid matching: Given a vocabulary of representative local descriptors \( P = \{p_k, k = 1, \ldots, K\} \) learned by k-means, the dense descriptor \( S = \{s_i, t = 1, \ldots, T\} \) at each point of the shape is replaced by the Gaussian kernel based soft assignment \( Q = \{q_k, k = 1, \ldots, K\} \).

Any function \( f \) on \( X \) can be written as the linear combination of the eigenfunctions. Using the variational characterizations of the eigenvalues in terms of the Rayleigh-Ritz quotient, the second eigenvalue is given by

\[
\lambda_2 = \inf_{f \neq 0} \frac{\int_{X} f C f}{\int_{X} A f f}
\]

The isocontours of the second eigenfunction (Figure 3) are used to cut the shape into \( R \) patches, thus the shape description is the concatenation of \( R \) sub-histograms of \( Q \) along eigenfunction value in the real line. To consider the two-sign possibilities in the concatenation, the histogram order is inverted, and the scheme with the minimum cost is considered as a better matching. The second eigenfunction is the smoothest mapping from the manifold to the real line, resulting in this intrinsic partition quite stable. It provably extends the property of popular SPM in image domain to capture spatial information for meshed surfaces, so is referred as intrinsic spatial pyramid matching (ISPM) in [LH13a]. The partition number is set as 2 in this contest.

Finally, the result is ISPM induced histograms for shape representation. The dissimilarity between two shapes is computed as the \( L_1 \) distance.

Running time The method is implemented in MATLAB. The time consuming steps of the method are the computation of LBO and k-means dictionary learning. For a mesh with 15,000 vertices, it takes 8 seconds to compute the LBO. For a mesh with 60,000 vertices, it takes 37 seconds to compute the LBO. To learn a dictionary with 100 geometric words, it takes 45 minutes. Therefore, it averagely takes at most 24 hours (less than one day) to run the program for each dataset.
5. Results

Here we evaluate the retrieval results of the methods described in Section 4, applied to the datasets described in Section 2. Table 1 shows the results of Task 1 evaluated using the NN, 1-T, 2-T, E-M and DCG measures discussed in Section 3. All methods performed better on the Synthetic dataset, with most methods performing poorly on the Real data. This shows that it is potentially easier to distinguish between synthetically generated objects, rather than objects captured from the ‘real world’, and that testing on synthetic data is not a reliable way to predict performance on real data.

The different classes in the Synthetic data may also be more easily distinguished because they have been manually designed to be different for this competition, whereas the models in the Real dataset were generated from body scans of human participants taken from an existing dataset, who may or may not have had very different body shapes. There is in fact a much higher similarity between the classes in the Real dataset. Figure 4 shows the precision-recall curve for the best performing methods submitted by each participant.

On the more challenging Real dataset, three methods, due to Litman et al., Ye, and Giachetti and Garro, performed significantly better than the others. The best performing method by Litman et al. was trained on a subset of the test set, and thus significantly better than the others. The methods used pre-existing knowledge of the size of each class. The organisers (Pickup et al.) submitted two very simple methods, Surface Area and Compactness. It is interesting to note that they perform better than many of the more sophisticated methods submitted, including their own, and that Surface Area is one of the top performing methods on the Synthetic dataset. These measures are obviously not novel, but they highlight that sophistication does not always lead to better performance, and a simpler and computationally very efficient algorithm may suffice. Algorithms should concentrate on what is truly invariant for each class.

Table 2 shows the results of Task 2 evaluated using the F-Measure. As for Task 1, the performance of all methods is much higher for the Synthetic dataset. All but one of the methods used pre-existing knowledge of the size of each class.

<table>
<thead>
<tr>
<th>Author</th>
<th>Method</th>
<th>NN</th>
<th>1-T</th>
<th>2-T</th>
<th>E-M</th>
<th>DCG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Giachetti</td>
<td>APT†</td>
<td>0.845</td>
<td>0.534</td>
<td>0.681</td>
<td>0.355</td>
<td>0.795</td>
</tr>
<tr>
<td>Lai</td>
<td>HKS-TS</td>
<td>0.235</td>
<td>0.259</td>
<td>0.461</td>
<td>0.314</td>
<td>0.548</td>
</tr>
<tr>
<td></td>
<td>SIHKS-H†</td>
<td>0.125</td>
<td>0.090</td>
<td>0.186</td>
<td>0.145</td>
<td>0.388</td>
</tr>
<tr>
<td>B. Li</td>
<td>Curvature</td>
<td>0.083</td>
<td>0.076</td>
<td>0.138</td>
<td>0.099</td>
<td>0.347</td>
</tr>
<tr>
<td></td>
<td>Geodesic</td>
<td>0.070</td>
<td>0.078</td>
<td>0.158</td>
<td>0.113</td>
<td>0.355</td>
</tr>
<tr>
<td></td>
<td>Hybrid†</td>
<td>0.045</td>
<td>0.080</td>
<td>0.164</td>
<td>0.117</td>
<td>0.354</td>
</tr>
<tr>
<td></td>
<td>Hybrid-R†</td>
<td>0.043</td>
<td>0.092</td>
<td>0.173</td>
<td>0.123</td>
<td>0.363</td>
</tr>
<tr>
<td></td>
<td>MDS-R</td>
<td>0.035</td>
<td>0.066</td>
<td>0.129</td>
<td>0.090</td>
<td>0.330</td>
</tr>
<tr>
<td></td>
<td>MDS-ZFDR</td>
<td>0.030</td>
<td>0.040</td>
<td>0.091</td>
<td>0.075</td>
<td>0.310</td>
</tr>
<tr>
<td>C. Li</td>
<td>Spectral Geom.†</td>
<td>0.313</td>
<td>0.206</td>
<td>0.323</td>
<td>0.192</td>
<td>0.488</td>
</tr>
<tr>
<td>Litman</td>
<td>supDLtrainR†</td>
<td>0.793</td>
<td>0.727</td>
<td>0.914</td>
<td>0.432</td>
<td>0.891</td>
</tr>
<tr>
<td></td>
<td>UnSup32</td>
<td>0.583</td>
<td>0.451</td>
<td>0.659</td>
<td>0.354</td>
<td>0.712</td>
</tr>
<tr>
<td></td>
<td>softVQ48</td>
<td>0.598</td>
<td>0.472</td>
<td>0.675</td>
<td>0.356</td>
<td>0.717</td>
</tr>
<tr>
<td>Pickup</td>
<td>Surface Area</td>
<td>0.263</td>
<td>0.289</td>
<td>0.509</td>
<td>0.326</td>
<td>0.571</td>
</tr>
<tr>
<td></td>
<td>Compactness</td>
<td>0.275</td>
<td>0.221</td>
<td>0.384</td>
<td>0.255</td>
<td>0.519</td>
</tr>
<tr>
<td></td>
<td>Canonical</td>
<td>0.010</td>
<td>0.012</td>
<td>0.040</td>
<td>0.043</td>
<td>0.279</td>
</tr>
<tr>
<td>Bu</td>
<td>3DDL</td>
<td>0.223</td>
<td>0.193</td>
<td>0.374</td>
<td>0.262</td>
<td>0.508</td>
</tr>
<tr>
<td>Tatsuruma</td>
<td>BoF-APFH</td>
<td>0.053</td>
<td>0.100</td>
<td>0.229</td>
<td>0.162</td>
<td>0.385</td>
</tr>
<tr>
<td></td>
<td>MR-BoF-APFH</td>
<td>0.048</td>
<td>0.071</td>
<td>0.131</td>
<td>0.084</td>
<td>0.327</td>
</tr>
<tr>
<td>Ye</td>
<td>R-BiHDM</td>
<td>0.275</td>
<td>0.201</td>
<td>0.334</td>
<td>0.217</td>
<td>0.492</td>
</tr>
<tr>
<td></td>
<td>R-BiHDM-s</td>
<td>0.685</td>
<td>0.541</td>
<td>0.742</td>
<td>0.387</td>
<td>0.781</td>
</tr>
</tbody>
</table>

Table 1: Retrieval results for Task 1. The 1st, 2nd and 3rd highest scores of each column are highlighted. † means the method has used part of the test data for training or parameter optimisation.

Real

Participant | Method | F-Measure | Synthetic
--- | --- | --- | ---
Giachetti | APT†‡ | 0.534 | 0.733
Lai | HKS-TS-HC†‡ | 0.063 | 0.244
| SIHKS-H-HC†‡ | 0.038 | 0.089
C. Li | Spectral Geometry‡ | 0.204 | 0.828
Litman | supDLtrainR | 0.640 | 0.814
Pickup | Surface Area‡ | 0.301 | 0.759
Bu | 3DDL‡ | 0.193 | 0.760

Table 2: Retrieval results for Task 2. The 1st, 2nd and 3rd highest scores of each column are highlighted. † signifies the method is aware of the class size, other annotation as for Table 1.

6. Conclusion

This paper compared non-rigid retrieval results obtained by 22 different methods, submitted by nine research groups, on two new datasets of human body models. These datasets are much more challenging than previous non-rigid datasets [LGB'11], as evidenced by lower success rates. The data obtained by scanning real human participants proved more challenging than the synthetically generated data. This shows that there is a lot of room for future research to improve the analysis of ‘real’ data. If the performance of methods is to be improved for real data, then more real datasets are needed for testing purposes, as synthetic datasets do not adequately mimic the same challenge.

All methods submitted were designed for generic non-rigid shape retrieval. Our new dataset has created the potential for new research into methods which specialise in shape retrieval of humans.

Acknowledgements

This work was supported by EPSRC Research Grant EP/J02211X/1.

References

[CJAM12] CORDEIRO DE AMORIM R., MIRKIN B.: Minkowski


