Class-specific 3D Localization using Constellations of Object Parts

Mukta Prasad[1]
mprasad@ee.ethz.ch

Jan Knopp[2]
jan.knopp@esat.kuleuven.be

Luc Van Gool[1]
vangool@vision.ee.ethz.ch

[1] ETH Zürich,
Zürich, Switzerland

[2] Katholic University of Leuven,
Leuven, Belgium

Abstract

Improvement in acquisition systems, has resulted in the ability to capture more realistic 3D models of real world objects, creating a need for better data processing techniques, as in the case of text and images. In this paper, we address the issue of learning class-specific, deformable, 3D part-based structure for object part localization in 3D models/scenes. We employ an inference framework upon fully connected part-based graphs inspired by Pictorial Structures (PS), which combine the local appearance of parts and the long-range structural properties. Using efficient tools for learning the model and performing inference, we show good results on a variety of classes, outperforming PS [7] and ISM [15]. Further, a similar inference framework is employed to find dense correspondences between 3D models, seeded by the above object part localization. Our results show promise for application in more complex 3D processing tasks such as part retrieval, pose estimation, scene understanding and recognition.

1 Introduction

The evolution of huge 3D repositories *e.g.* google warehouse [1], and systems like Kinect [25], has boosted the need for effective 3D scene understanding, pose-estimation, localization, registration *etc.* The 3D shape localization task is related to various strands of literature: 2D recognition and localization, 3D shape representation and shape matching. Fischler and Elschlager [9] proposed representing objects as a collection of parts joined by "virtual" springs. The appearance of each object part is modelled locally, while the joint locations of object part detections must satisfy the long-range characteristics of the model. Felzenswalb and Huttenlocher [7] revived this idea, reducing the graph (of all possible connections between object parts) to a tree, on which inference could be performed exactly and efficiently. In a parallel effort, Fergus *et. al.* [8], showed how the more complex, fully-connected graph could be approximately inferred using EM-based techniques.

This paper describes a method for localizing 3D parts using a constellation-based approach and finding dense correspondences initialized from the result. While 2D methods matured gradually to issues of handling orientation, scale and deformation, in 3D these must be handled outright. 3D models are often either too smooth (handmade) or too noisy (from

BMVC 2011 http://dx.doi.org/10.5244/C.25.34

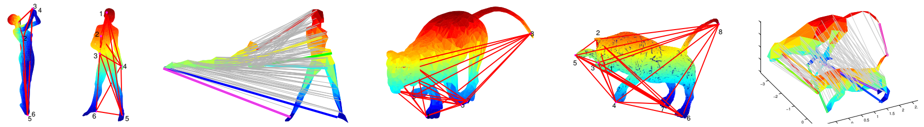


Figure 1: **First look:** Fully connected graphs of object part nodes are learnt and inferred for 2 models each of classes “Victoria” and “Lion” from the TOSCA dataset. Subsequently, correspondences can be estimated between the 3D models. Note: Correspondences are depicted on a subsampled resolution of the hierarchical optimization, for ease of viewing. Note: the models are coloured arbitrarily for 3D visualization.

range-scans). Most databases lack the colour information taken for granted in 2D. The problem of finding “good” features for 3D–global (SH [13], SD [20] *etc.*) and local (SI [11], HKS [26], FPFH [24])–has been discussed at length. What’s more, even popular 2D features such as SURF have been extended to 3D [15]. (While no one feature combines the advantages of all, the reliability of these has allowed people to move to the next step of retrieval (BOF based [21]), classification ([28]) and recognition([2])).

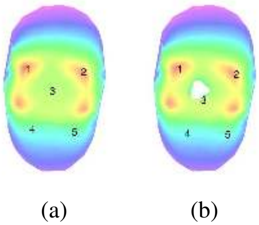
For pose estimation, the problems of orientation, noise, deformation and finding good shape description, means that performing ICP-based 3D matching on huge databases, is very expensive and prone to minima. Knopp *et. al.* [15] showed that local and global shape properties could be combined with the Hough voting based scheme of implicit shape models (ISM [18]) for object localization. Kinect [25] has used state-of-the-art learning and hardware to perform real-time pose-understanding for the case of humans.

For many applications involving 3D data, effective recognition, matching and retrieval play an important role. In the presence of holes or noise, part-based matching can clearly suffer from information loss. Differences in scale and cross-class information can also lead to difficulties. Additionally, if the part that we are trying to match is in a low detail area, then matching to relevant parts can be tricky. In this context, learning the structure of objects can be useful. *E.g.* Having observed the head and three paws of a cat, we should be in a better position to predict what is missing (say the fourth paw), and where it should be, despite model noise/holes. Knowing this, should enable us to search more intelligently and initialize matching from methods such as ICP better. In 2D, the principled and effective framework of pictorial structures has made many researchers return to it. It is natural to explore its relevance for 3D localization tasks.

Overview: We describe the Pictorial Structures (PS) method for 3D [7] in § 2. The setup is described in § 2.1 extended the fully connected (FC) approach in § 2.2. We compare our method to PS in § 3.2 and the Hough-voting based ISM method in § 3.3. The localization results are used to find good, dense correspondences between deformed class instances § 3.4. We finally summarize our findings in § 4.

2 Pictorial Structures

We are given a set of 3D shape instances $\{S_m | m \in \{1 \cdots M\}\}$, denoted as a set of vertices $\{\underset{3\times 1}{v_{mv}} | v \in \{1 \dots V_m\}\}$ and edges $\{\underset{2\times 1}{\varepsilon_{me}} | e \in \{1 \dots E_m\}, \varepsilon_{mjk} \in \{1 \dots V_m\}\}$. The appearance of the shape at the v^{th} vertex is described by a descriptor $\mathbf{s}_{mv}$. In this work, we use orientation invariant Heat Kernel Signature descriptors for this (HKS [21]). Given P parts located at L_{mp} for each S_m, the aim is to learn a graphical model (governed by parameter set Θ), in an inference framework. The object parts, governed by appearance parameters $\alpha = \{\alpha_p | p \in \{1 \dots P\}\}$,



(a) (b)

Figure 2: **Class-specific part-based graph:** (a) User-marked salient black points plotted over face model. The colour encodes distance of surface HKS features from salient point 1. (b) The same face in the "hole" test dataset. The variation of features from salient pt. 1 is similar. Despite the hole, the part locations inferred in (b) are similar to the ground truth of (a).

define the nodes of a graph $G \subset F$, (F: fully connected graph over P parts/nodes), whose edge parameters are $\gamma = \left\{\gamma_{ij} | G_{ij} \in G\right\}$. $\Theta = \{G, \alpha, \gamma\}$ must be learnt from the training shapes and configurations. The learnt model is then used to jointly estimate (or sample) the optimal layout of object parts L^* in a query shape S^* (or scene) by maximizing the posterior as shown.

$$L^* = \underset{L}{\operatorname{argmax}}\, p(L|S^*, \Theta) = \underset{L}{\operatorname{argmax}}\, p(S^*|L, \Theta) p(L|\Theta) \approx \underset{L}{\operatorname{argmax}}\, p(S^*|L, \alpha) p(L|\gamma, G) \quad (1)$$

$$\text{where, } p(S^*|L, \alpha) = \prod_{p=1}^{P} p(\mathbf{s}_{L_p}|\alpha_p) \quad (2)$$

$$p(L|\gamma, G) \propto \prod_{\{i,j\} \in G} \psi(L_i, L_j; \gamma_{ij}) \quad (3)$$

Ideally, Θ is learnt by Maximum-Likelihood estimation from the training data. Assuming that the parameters α and (G, γ) independently influence the part-appearance likelihoods and the pairwise-part configuration likelihoods respectively, the joint probability is approximated across the M instances to learn Θ as:

$$\Theta = \underset{\Theta}{\operatorname{argmax}} \prod_{m=1}^{M} p(S_m, L_m|\Theta) = \underset{\Theta}{\operatorname{argmax}} \prod_{m=1}^{M} p(S_m|L_m, \alpha) \prod_{m} p(L_m|\gamma, G) \quad (4)$$

For a tree-structured G, the parameters can be separated and learnt efficiently using the following:

$$\alpha_p = \underset{\alpha_p}{\operatorname{argmax}} \prod_{m} p(\mathbf{s}_{mL_{mp}}|\alpha_p), \quad (5)$$

$$G = \underset{G \subset F}{\operatorname{argmax}} \prod_{m, \{i,j\} \in G} \psi(L_{mi}, L_{mj}; \gamma_{ij}), \quad (6)$$

$$\gamma_{ij} = \underset{\gamma_{ij}}{\operatorname{argmax}} \prod_{m} \psi(L_{mi}, L_{mj}; \gamma_{ij}) \text{ for } \{i, j\} \in G. \quad (7)$$

In [7], optimizing (5) and (7) is equivalent to fitting distributions (gaussians) on the training data, which can be learnt independently for the nodes and edges of the graph. To make inference simple and optimal, [7] simplifies graph G as the minimum spanning tree over edge costs defined on F, implicitly helping us identify which edges are the most 'informative' (see [7] for algorithmic and implementation details).

2.1 Application to 3D data

Having introduced PS, we now see how it is applied to 3D scenes. Given the information in § 2, the parameters α_p of a distribution can be learnt by optimizing (5). A normal distribution $\mathcal{N}(\alpha_{\mu_p}, \alpha_{\Sigma_p})$ is used here (in the absence of enough training samples, α_{Σ_p} is often further restricted to be diagonal). Similarly, the likelihood of a pair of nodes occurring at two locations on the 3D shape is a function of their mutual locations. For invariance to similarity transforms (scale, rotation, translation), the relative distance between nodes i and j is computed, normalized to the global object scale. Directed or absolute euclidean distance is usually used in the 2D scenario. We compare euclidean and geodesic measures for distance in (9); the latter demonstrates greater invariance to object deformation, and stability to intra-class variation and articulation. γ is learnt similarly as α, after extracting the minimum spanning tree G (see [7]). The unary and pairwise terms of (5, 7) can be defined as energies/costs for a given shape S_m as:

$$p(\mathbf{s}_{mL_p}|\alpha_p) \propto \exp\left(-\phi(\mathbf{s}_{mL_p};\alpha_p)\right) \propto \exp\left(-0.5(\mathbf{s}_{mL_p}-\alpha_{\mu_p})^\top \alpha_{\Sigma_p}^{-1}(\mathbf{s}_{mL_p}-\alpha_{\mu_p})\right), \tag{8}$$

$$\psi(L_{mi},L_{mj};\gamma_{ij},\{ij\}\in G) \propto \exp\left(-0.5(\mathrm{dist}(L_{mi},L_{mj})-\gamma_{\mu_{ij}})^\top \gamma_{\Sigma_{ij}}^{-1}(\mathrm{dist}(L_{mi},L_{mj})-\gamma_{\mu_{ij}})\right). \tag{9}$$

The pairwise energy (9) differs slightly from the standard energy used by [7, 8]. The location of the p^{th} object part: L_{mp}, spans the object surface, discretized to the set of labels/possible vertices $v = \{1 \dots V_m\}$. The likelihood of a given surface location $L_{mp} = v$ being a specific object part p is computed using (8). The geodesic cost of assigning part locations L_i, L_j to labels/vertices i, j in (9) is approximated by the cost of the shortest path $\mathrm{dist}(i, j) = \mathrm{dijkstra}(i, j)$ given $S = \{\nu, \varepsilon\}$ between i, j.

Given the energies for the unary and pairwise terms, estimation of the part configuration for a query shape can be computed by existing tools for inference (BP [23], TRW-S [16]). Inference could be done to either yield a posterior distribution which can be sampled for likely configurations, or to simply find the maximum a-posteriori (MAP) estimate (the "best" configuration). We do the latter. For the tree-based graph G, inference is exact and fast.

2.2 Inference on a fully connected (FC) graph

Depending on the exact form of the tree based graph, some nodes (especially leaf nodes) which are not directly connected, lack repulsion to each other. *E.g.* when trying to fit the head and paws of a cat to the query model of a cat for a tree learnt as in fig. 3(b), nothing prevents two model paws from converging on one in the test model. This problem is fairly well discussed (*e.g.* [17]). Thus, the need for more complex graphs *e.g.* a fully-connected (FC) graph (the original F in § 2, also see fig. 3(c)). This is similar to the constellation model proposed by Fergus *et. al.* [8], but solved with better, recent inference tools instead of an EM-based optimization, for the supervised case. In the choice between the optimal tree-based and sub-optimal FC case, [7] chose the former, as global optimization is much easier in the 2D case. However, inference tools such as [16] enable an efficient attempt at approximate inference while evaluating how close we are to the global optimum, for such loopy graphs in 3D. Therefore, we upgrade to the FC graph $G = F$. The unaries and pairwise terms stay as before (8,9). Exact training of the FC graph is intractable. Here, we approximate them as in (5-7). The estimation is performed as before using TRW-S [16] on the loopy F. The important result here, is that approximate inference over the more complicated FC model yields better performance than exact optimization on the simpler PS one. The experimental setup and findings are detailed now.

3 Experiments

Part localization in objects and object localization in scenes is essential for tasks of recognition and retrieval in object datasets. We now show our evaluation of the basic PS set up in § 3.1 and show the improvement of our fully connected (FC) model over PS in § 3.2. We then compare our method against another established method of object detection: Hough-voting based ISM in § 3.3. We then show an extension of our method for the purpose of dense correspondence establishment in § 3.4.

3.1 Dataset

The experiment was conducted on 10 object classes (cat, centaur, david, dog, face, gorilla, horse, lioness, michael, victoria) from the TOSCA dataset [5] (each dataset has between 15-20 models, split in half for training and testing, each model with ~ 7000 vertices and ~ 3000 faces). The TOSCA 3D models have relatively little noise, but good surface detail, mesh quality and realistic object deformations. Ideally, we would like the system to be totally unsupervised. However, the task of finding salient points that occur consistently across all models (through clustering and salient feature detection) was difficult. Therefore, for training, the user marks a few $(5-8)$ salient points (*e.g.* eyes, mouth, paws, tail) across instances of each class, which is a relatively simple job. These points define object parts/nodes in the graph. Additionally, we construct an artificial "hole dataset", by creating holes in the above TOSCA test set at the locations of each of the part positions (one part at a time) for each class (see fig. 2). The hole, while removing the object part detail, only affects the object model by $5-10\%$. As described in § 2.1, HKS descriptors are calculated on all the models including the hole dataset. We found HKS to be much more stable and proportional to changes on the surface than other feature types. An example of the test dataset, hole dataset, annotation and smooth feature variation is illustrated in fig. 2. For performance evaluation of algorithms, accuracy is measured with respect to the user-marked ground-truth annotations on test data. Interestingly, classes such as cats are symmetric around a plane passing through the middle of their body. Therefore at test time, both allowable flips are taken into consideration while evaluating error. Another dataset of interest is the SHREC correspondence dataset (see [4]). This dataset has a variety of transformations, noise and holes added to it to aid in benchmarking. It has three classes of objects: human, dog, horse. We will use our learnt object part models to evaluate against it qualitatively.

3.2 PS vs. FC

As described in § 2.1-2.2, the training parameters are learnt. At test time, inference over a query involves finding a configuration that maximizes the combination of unary (8) and pairwise (9) terms. The inference problem involves finding the optimal configuration or the optimal placement of nodes in the model/scene. This is equivalent to a labelling problem, where each node of the graph (object part) can take one of the N possible labels (~ 7000 vertex positions). In turn, each edge of the graph can have $N \times N$ combinations of node labels on its ends. Optimization involves enumerating the unaries and pairwise terms for all such labellings. For unaries (8) this is easy. The distance between any two labels, or any two positions on the model graph can be found (9) effectively using the Floyd-Warshall algorithm. Inference is then performed for an MAP estimate. The current optimum can also be compared to the global possible optimum to evaluate the quality of the optimization.

We run the tree-based PS model of [7] on test dataset described and the performance are summarized in (see fig. 3(b), table 1 (col 4)). We then evaluate our fully connected FC model.

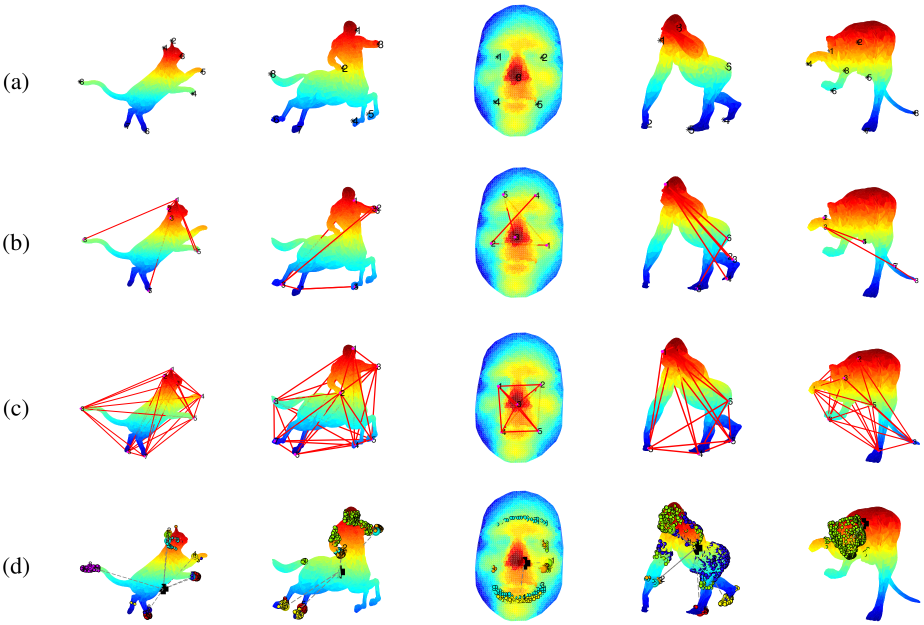


Figure 3: **3D Shape localization:** We compare the results of running PS of [7] (b) with our FC model (c) and the ISM model (d) of Knopp *et. al.*[15] against the ground truth (a). Our results outperform the others, considerably. The ground truth can be noted in (a). In (d), the valid contributing features to the ISM hypothesis are marked in blobs (coloured uniquely by part). The best part locations are connected to the estimated object centre (denoted by a +). While the detected parts are usually in the correct area, many go missing when the object undergoes deformation.

The results are seen in figs. (3(c), 5) and table 1. The difficulty of encouraging repulsion amongst the nodes is hardly exaggerated, as can be seen in the superior performance both in the numbers and the quality of the output. Therefore, despite a much more complex model, we find our inference mechanism is consistently superior to the tree-based PS approach. Despite not reaching the best solution, the TRW-S solution tells us how far we are from the lower bound, which is also informative.

3.3 ISM vs FC

We evaluate our findings by comparing our FC model with the popular hough-voting based ISM framework of [18] used for 3D localization in Knopp *et. al.* [15], in a framework equivalent to ours. In its original form, the voting based algorithm clusters visual words and uses the elements of each cluster to vote for the object center location using memory from the visual word occurrences stored from training. In our experiment, we use the user-clicked salient points as the training visual word occurrences. Book-keeping from training data helps to retrieve relative locations to the object centre which is used to cast votes for the object centre. The rotation invariant voting methods of [14] were used for handling orientation ambiguity in the models. The voting space is searched for a maximum. This is used to validate those features that voted correctly for it. The means of the validated features belong to each visual word, then form the detection of that visual word. In [15], ISM voting is done in a euclidean framework while our framework optimizes geodesic distances so errors are compared in both measures. The ISM results for the complete and hole test datasets are

	I. euclid			II. geodist					I.Euc,hole			II.geodist,hole			
	FC	ISM	ISM (p)	PS	FC	ISM	ISM (p)	% ISM	FC	ISM	ISM(p)	FC	ISM	ISM(p)	% ISM
cat	0.13	0.50	3.61	1.23	0.15	0.57	4.15	5.75/8	0.17	0.63	3.05	0.25	1.42	3.84	5.26/8
centaur	0.43	0.57	1.25	1.25	1.01	1.09	1.57	3.33/8	0.41	0.72	1.21	0.89	2.92	2.34	4.21/8
david	0.75	0.95	1.36	2.33	1.39	1.47	1.87	2.75/5	0.78	1.03	1.37	1.60	2.39	2.34	2.78/5
dog	0.22	0.78	1.12	1.26	0.54	1.22	1.44	5.33/8	0.24	0.84	1.15	0.41	1.23	1.44	5.33/8
face	0.1	0.71	0.78	0.29	0.10	0.86	0.92	4.63/5	0.54	0.81	0.91	0.60	1.09	1.16	4.48/5
gorilla	0.65	1.27	1.33	3.08	1.11	2.42	2.47	4.91/6	0.69	1.35	1.41	1.14	2.93	2.82	4.74/6
horse	0.38	0.96	1.53	1.50	0.66	1.21	1.67	1.78/8	0.40	1.31	1.61	0.67	2.29	2.06	1.83/8
lioness	0.51	0.47	2.84	1.57	0.83	0.84	3.88	3.86/8	0.54	0.51	2.62	0.96	1.62	3.83	3.98/8
michael	0.62	0.75	1.09	2.42	1.18	0.97	1.43	4.2/6	0.61	0.83	1.11	1.28	2.24	2.07	4.37/6
victoria	0.65	0.92	3.03	2.97	1.51	1.37	5.01	2.3/6	0.67	0.92	3.00	1.35	1.43	4.99	2.5/6
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Table 1: **Numerical Evaluation**: Our FC model is compared with the best case ISM method. The full and "hole" versions of the TOSCA class datasets are explored. Both euclidean and geodesic distance based errors of prediction are evaluated on unit-scale models *w.r.t.* ground truth. Note, these measures lie in two different spaces and cannot be directly compared. ISM often results in partial matches, therefore the naive ISM error is stated alongside the ISM error penalized for missing parts. For explanation please see § 3.3.

summarized in fig. 3(d) and table 1.

In ISM, the mean of these feature locations belonging to a visual word doesn't have to lie on the model and is thus mostly incorrect with respect to the ground truth. At this level, our FC method (table 1 cols 1,9) wins hands down with respect to ISM (cols 2,10). We give ISM additional benefit; instead of choosing the mean of the visual word detections, we choose the best possible surface features of this visual word as its object part location. Now the geodesic error disparity of ISM (cols 6,13) understandably reduces *w.r.t* us (cols 5,12). Also, the validation step of the ISM acts as a filter which removes those features that don't vote for the final object centre. This pruning is aggressive and often results in the removal of many parts (fig. 3(d): some parts are detected correctly, but many are left out). The error of this model is therefore measure with (cols 2,6,10,13) and without penalizing (cols 3,7,11,14) missing parts. While the average un-penalized error is low (in the best cases of cols 6,13) implying few false positive part detections, the localizations are clearly inadequate (see fig. 3). For the penalized version, the penalty of each missing part is set at the average distance of model vertices in euclidean and geodesic distance measures. This causes an increase the ISM error (cols 3,7,11,14) as is expected when parts go undetected in accurate models. The percentage of the nodes that are detected by ISM across the test classes are also shown in col. 15.

As can be seen, our approach generally outperforms the ISM results both in the penalized and un-penalized versions. In the hole dataset, these differences are heightened. One could argue that the user-annotated ground truth is "slightly faulty" in itself as there is really no one location where "the foot" is in the space of foot-related vertices. In this context the visual quality of our results speak for themselves (see fig. 3). The ISM based framework approximates a star-formation graphical model. However, unlike other methods inference here is done approximately and alternately (in a 2 step iteration). The voting space is discretized and many parameters affect the performance of this system. Therefore, while the error in individual part localizations is not so high, relevant parts are left out of the hypothesis far too easily. While in the presence of holes this may be a good thing, in most full, noiseless, complete models this is definitely a mistake. The advantage of inference on the fully connected model is that it performs a joint optimization on the objective, with very few parameters to hand-tune.

3.4 Dense correspondence estimation

Correspondence estimation (sparse and dense) is an important problem for many applications such a motion capture, morphing *etc.* The simplest way of solving such an assignment

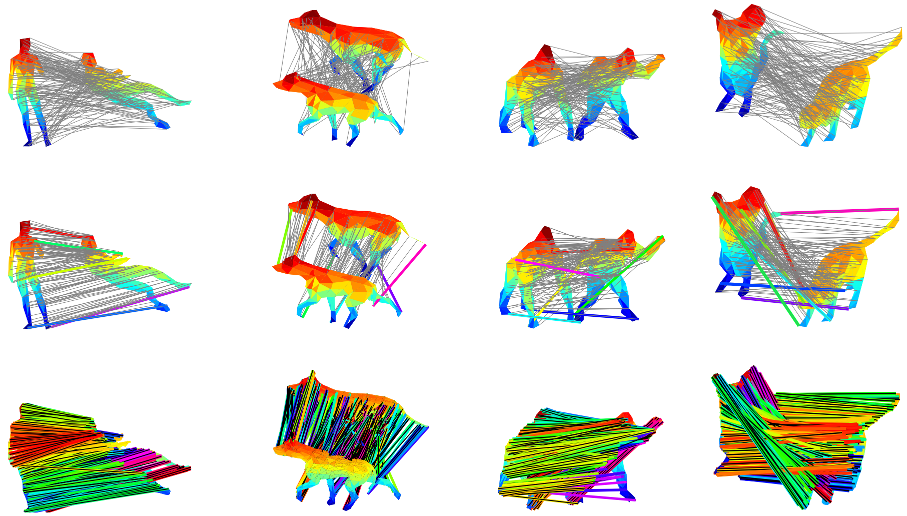


Figure 4: **Correspondence finding:** The results of the naive Hungarian algorithm (row 1) is compared with our method at 300 vertices (row 2) and then at 1200 vertices (row 3).

problem is employing the Hungarian algorithm for bipartite graph matching (similar to [19]). More exact/interesting algorithms have been applied in 2D *e.g.* [3],[6], [31],[4] and some in 3D ([10],[5],[27],[12],[30]).

The Hungarian algorithm uses the similarities of nodes in a graph for assignment, but now we're also interested in spatial consistency, *i.e.* every vertex i of a 3D shape S_m should be matched to a vertex L_i of S_n, such that similar parts are in correspondence, while preserving spatial smoothness of assignments. We extend the idea of § 2; the graph G is replaced with vertex nodes ν_m and edge connectivities ε_m (as in § 2.1) of shape S_m. Inference of configuration L_m is now replaced with the task of assigning the appropriate vertex label $L_i \in \{1 \dots V_n\}$ of S_n to each vertex $i \in \{1 \dots V_m\}$ of S_m. Similar to (8), the unary potential evaluates how well the shape S_n at vertex L_i described by $\mathbf{s}_{nL_i}$ matches $\mathbf{s}_{mi}$. The pairwise term now encourages labels $L_i, L_{i'}$ in S_n to have a similar relative distance to that of neighbouring nodes i, i' in S_m. Finding and localizing object part structure as in § 2-2.2, gives us a sparse correspondence of salient object parts, which can be encoded as hard constraints to seed the dense correspondence optimization. Part p localized to i in S_m and to k in S_m, gives us a vertex correspondence i, k between them. In summary, the unary and pairwise potentials can be written as:

$$\phi(L_i;\lambda) = \begin{cases} \infty, & (L_i \neq k)\,\&\,(i,k \text{in corr.}) \\ \lambda(\mathbf{s}_{nL_i} - \mathbf{s}_{mi})^2, & \text{otherwise} \end{cases} \tag{10}$$

$$-\log(\psi(L_i, L_{i'}|\lambda)) = \begin{cases} \infty, & (L_i \neq k)\,\&\,(i,k \text{in corr.}) \\ (\text{dist}_n(L_i, L_{i'}) - \text{dist}_m(i,i'))^2, & \text{otherwise} \end{cases} \tag{11}$$

Experiment: The above energy can be solved by applying TRW-S as in § 2.2. λ is empirically set. In our problem, with meshes of ~ 3000 nodes and label range of ~ 3000, this is a computationally daunting task. To get around this, we perform this correspondence establishment hierarchically, the mesh resolution is increased in steps of ~ 300 vertices. The shape descriptors from the highest resolution are retained and at any resolution the descrip-

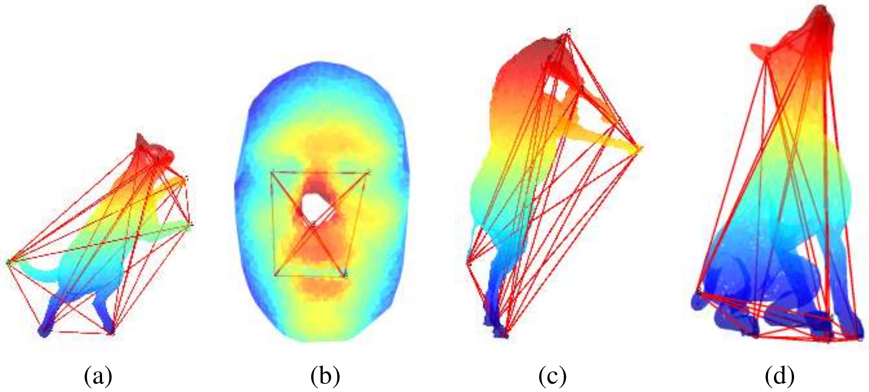


Figure 5: Some results on TOSCA hole and SHREC test datasets. (a,b) Shapes from the TOSCA dataset with holes (missing faces). (c,d) Examples from SHREC'10 dataset with provided deformations such as artificial noise, micro-holes, local scale change or normal holes.

tors to the closest high-resolution vertex are employed. The result of the previous stage is used as fixed correspondences in the subsequent stage. Labels that are highly unlikely may even be pruned. Thus a relatively simple problem is solved at each step. Naturally, the results of the first few stages are crucial in determining the outcome. The results of performing this experiment can be seen in fig. 4 compared to a simple Hungarian matching baseline.

4 Summary

We have shown how graph based inference can be used to learn object part structure similar to the popular work of Pictorial Structures, by extending the algorithm to 3D, for object part localization and dense correspondence finding. We further explore the power of fully connected object part graphs with pairwise cliques to allow the model maximum power. We considerably outperform Pictorial Structure and ISM based benchmarks, both quantitatively and qualitatively. The ability to learn object part structure affords us the opportunity to perform more complex tasks in 3D such as semantic in-painting of missing parts, object classification, morphing, correspondence establishment for motion capture *etc.* Going forward, it would also be useful to study the sensitivity of the localization results to the object parts chosen by the user. Future extensions of this work include removing supervision, other methods of training [22] and inference [29] and examining the method's robustness to factors such as user annotation error, meshing quality *etc.*

5 Acknowledgements

We thank Thomas Minka for critical insight and proof-reading. This work is funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 231809 (IP project "3D-COFORM").

References

[1] http://sketchup.google.com/3dwarehouse/.

[2] P. Bariya and K. Nishino. Scale-hierarchical 3d object recognition in cluttered scenes. In *Proc. CVPR*, 2010.

[3] S. Belongie and J. Malik. Shape matching and object recognition using shape contexts. *IEEE PAMI*, 24(24), 2002.

[4] A. Bronstein, M. Bronstein, B. Bustos, U. Castellani, M. Crisani, B. Falcidieno, L. Guibas, I. Kokkinos, V. Murino, I. Sipiran, M. Ovsjanikovy, G. Patanè, M. Spagnuolo, and J. Sun. Shrec 2010: robust feature detection and description benchmark. In *3DOR'10*, pages 79–86. Eurographics Association, 2010.

[5] M. Bronstein, A. Bronstein and R. Kimmel. *Numerical Geometry of Non-Rigid Shapes*. Springer Publishing Company, Incorporated, 2008.

[6] T. S. Caetano, L. Cheng, Q. V. Le, and A. J. Smola. Learning graph matching. In *Proc. ICCV*, 2007.

[7] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1): 55–79, 2005.

[8] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. CVPR*, volume 2, pages 264–271, Jun 2003. URL http://www.robots.ox.ac.uk/~vgg.

[9] M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, c-22(1):67–92, Jan 1973.

[10] R. Gal and D. Cohen-Or. Salient geometric features for partial shape matching and similarity. *ACM Trans. Graph.*, 25(1):130–150, 2006.

[11] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE PAMI*, 21(5):433–449, 1999.

[12] E. Kalogerakis, A. Hertzmann, and K. Singh. Learning 3D Mesh Segmentation and Labeling. *ACM Transactions on Graphics*, 29(3), 2010.

[13] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Rotation invariant spherical harmonic representation of 3d shape descriptors. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, SGP '03, pages 156–164, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[14] J. Knopp, M. Prasad, and L. Van Gool. Orientation invariant 3D object classification using hough transform based methods. In *ACM Multimedia 2010 Workshop - 3D Object Retrieval*, 2010.

[15] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. Van Gool. Hough transform and 3d surf for robust three dimensional classification. In *Proc. ECCV*, 2010.

[16] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE PAMI*, 28(10):1568 –1583, Oct 2006. ISSN 0162-8828. doi: 10.1109/TPAMI.2006.200.

[17] X. Lan and D.P. Huttenlocher. Beyond trees: common-factor models for 2d human pose recovery. In *Proc. ICCV*, volume 1, pages 470–477, Oct 2005.

[18] B. Leibe and B. Schiele. Interleaved object categorization and segmentation. In *Proc. BMVC.*, volume 2, pages 264–271, 2003.

[19] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *Proc. ICCV*, 2005.

[20] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobki. Shape distributions. *ACM Transactions on Graphics*, pages 807–832, 2002.

[21] M. Ovsjanikov, A.M. Bronstein, M.M. Bronstein, and L.J. Guibas. Shape google: a computer vision approach to isometry invariant shape retrieval. In *NORDIA*, 2009.

[22] S. Parise and M. Welling. *Learning in Markov Random Fields: An Empirical Study*. 2005.

[23] J. Pearl. *Probabilistic Reasoning in Intelligent Systems : Networks of Plausible Inference*. Morgan Kauffman, San Mateo, California, 1988.

[24] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (FPFH) for 3D registration. In *Proc. Intl. Conf. on Robotics and Automation*, pages 3212 –3217, may 2009.

[25] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. Technical report, Microsoft Research Cambridge, Xbox Incubation, 2010.

[26] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *SGP*, pages 1383–1392, 2009.

[27] H. Tabia, M. Daoudi, J. P. Vandeborre, and O. Colot. A new 3d-matching method of nonrigid and partially similar models using curve analysis. *IEEE PAMI*, 33:852–858, April 2011.

[28] R. Toldo, U. Castellani, and A. Fusiello. A bag of words approach for 3d object categorization. In *MIRAGE*, 2009.

[29] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization. In *Proc. ECCV*, pages 596–609, Berlin, Heidelberg, 2008. Springer-Verlag.

[30] O. Van Kaick, H. Zhang, G. Hamarneh, and D. Cohen-Or. A survey on shape correspondence. *Computer Graphics Forum (CGF) (extended version of Eurographics 2010 STAR)*, pages (in press, 23 pages), 2011.

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