On the Use of Implicit Shape Models for Recognition of Object Categories in 3D Data

Samuele Salti, Federico Tombari, and Luigi Di Stefano

Computer Vision Lab, DEIS, University of Bologna, Bologna, Italy
{samuele.salti,federico.tombari,luigi.distefano}@unibo.it
www.vision.deis.unibo.it

Abstract. The ability of recognizing object categories in 3D data is still an underdeveloped topic. This paper investigates on adopting Implicit Shape Models (ISMs) for 3D categorization, that, differently from current approaches, include also information on the geometrical structure of each object category. ISMs have been originally proposed for recognition and localization of categories in cluttered images. Modifications to allow for a correct deployment for 3D data are discussed. Moreover, we propose modifications to three design points within the structure of a standard ISM to enhance its effectiveness for the categorization of databases entries, either 3D or 2D: namely, codebook size and composition, codeword activation strategy and vote weight strategy. Experimental results on two standard 3D datasets allow us to discuss the positive impact of the proposed modifications as well as to show the performance in recognition accuracy yielded by our approach compared to the state of the art.

1 Introduction

Object categorization is among the most stimulating, yet challenging, computer vision tasks. It consists of automatically assigning a category to a particular object given its representation (an image, a point cloud, ..) and a predefined taxonomy. In the last decade the main effort has been devoted to categorizing classes of objects from images [1], one of the most prominent approaches being the application to image features of the Bag-of-Words paradigm, previously used for text categorization and document analysis. In particular, this approach, typically referred to as *Bag-of-Features* (BoF) or *Bag-of-Visual-Words* (BoVW), represents image categories as histograms ("bags") of feature descriptors [2,3,4]. To account for efficiency, histograms are not built on descriptors themselves but on an alphabet of descriptors, typically termed "codebook", obtained via clustering or vector quantization [1].

BoF methods turned out to be particularly effective even though, unlike some more recent proposals, they completely discard geometrical relationships between object parts. Among those leveraging geometric structure, one of the most successful proposals is Implicit Shape Models (ISM) [5], that encodes spatial relationships by means of a probabilistic Hough voting in a 3-dimensional

R. Kimmel, R. Klette, and A. Sugimoto (Eds.): ACCV 2010, Part III, LNCS 6494, pp. 653–666, 2011.

space representing scale and translation. Moreover, the use of geometrically well-localized information allows these methods to be deployed also as detectors of specific object categories in presence of clutter, occlusion and multiple object instances. Typical object categories of interest have been pedestrians, faces, humans, cars [5].

The increasing availability of large databases of 3D models has pushed forward a growing interest towards computer vision and machine learning techniques capable of processing 3D point clouds and meshes. One of the most investigated tasks so far has been 3D object retrieval (see [6,7] for surveys) which aims at finding the most similar 3D models in the database to a given query model inputted by the user. Another well investigated topic concerns 3D object recognition [8,9]. Only very recently the first methods aimed at 3D object categorization have been proposed in literature. They mainly extend the BoF paradigm to the 3D scenario by representing categories as histograms of codewords obtained from local shape descriptions of 3D features [10, 11, 12].

In this paper we investigate on how to deploy Implicit Shape Modeling for the categorization of 3D data. Although in the reminder of this paper we will focus only on categorization, it is worth noting that this approach holds the potential to solve within the same framework the problem of simultaneous localization and classification of objects in cluttered scenes, even in presence of multiple instances.

2 3D Implicit Shape Model

The basic idea idea underlying Implicit Shape Models is to perform object category recognition and instances localization based on a non-parametric probability mass function of the position of the object center. These probability functions come from a probabilistic interpretation of the voting space of a Generalized Hough Transform algorithm. Votes are casted by local features that are matched against a codebook learned, together with votes, from a set of training examples. When applied to 3D data, we identify the general form of an algorithm training a 3D ISM as follows (Fig. 1):

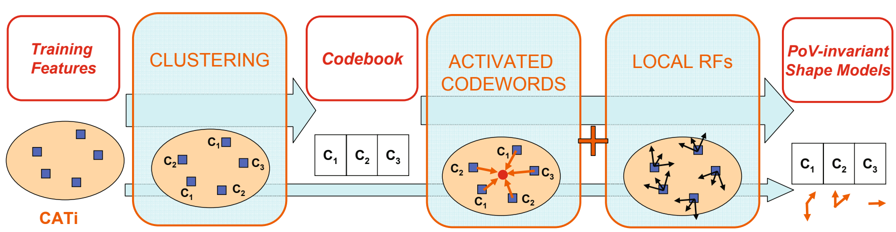


Fig. 1. Overview of the training stage of 3D ISM

– local features are detected and described from the 3D training data.
– for each category C_i
  • all features belonging to C_i are clustered to create the codebook of C_i
  • for each training feature $f_j^{C_i}$ of category C_i
    * $f_j^{C_i}$ is matched against the codebook of C_i according to a *codeword activation strategy*.
    * each activated codeword adds to the ISM of C_i the position of $f_j^{C_i}$ with respect to the object center. Each feature $f_j^{C_i}$ needs to incorporate a repeatable local Reference Frame (RF), and votes are expressed with respect to such local RF of $f_j^{C_i}$.

Then, a generical 3D ISM recognition procedure may be decomposed in the following steps (Fig. 2):

– local features are extracted and described from the 3D input data.
– for each feature f_j and each category C_i
  • f_j is matched against the codebook of C_i according to a *codeword activation strategy*.
  • each activated codeword casts its set of votes for the Hough Space of C_i in its ISM.
  • votes are rotated and translated so as to be expressed in the local RF of the input features before voting, thus obtaining *Point-of-View (PoV) independent votes*. The magnitude of the vote is set according to a *vote weighting strategy*.
– in case of categorization of 3D database entries, the category yielding the global maximum among all the Hough spaces is selected as output; in case of detection in a cluttered scene, local maxima of each category above a threshold are selected as category instance hypotheses for a further verification stage and/or pose estimation.

This scheme exhibits two main differences with respect to the use of ISM for detection of object categories in 2D images. First of all, since the sensor produces metric data, there is no need for scale invariance: in the 2D case, when casting votes for the object center, the object scale is treated as a third dimension in the voting space. With 3D data we can cast votes for object hypotheses directly in the coordinates space, which is again a 3D dimensional space. The second difference regards the use of PoV-independent votes, that leads to a PoV-independent detector. In the original ISM proposal, objects of the same category being seen under different point of views are regarded as instances of different, unrelated categories. It is worth pointing out that the use of PoV-independent votes is not just a nice extension that allows for more flexibility of the final method, it is indeed mandatory when using 3D ISM to categorizes 3D database entries, for these cannot be assumed to be expressed within the same global RF. Unfortunately, most of the proposals in the field of 3D local features do not include a fully defined local RF, *e.g.* Spin Image [8] uses just one repeatable axis, the normal, and 3D Shape Context [9] uses a random, not

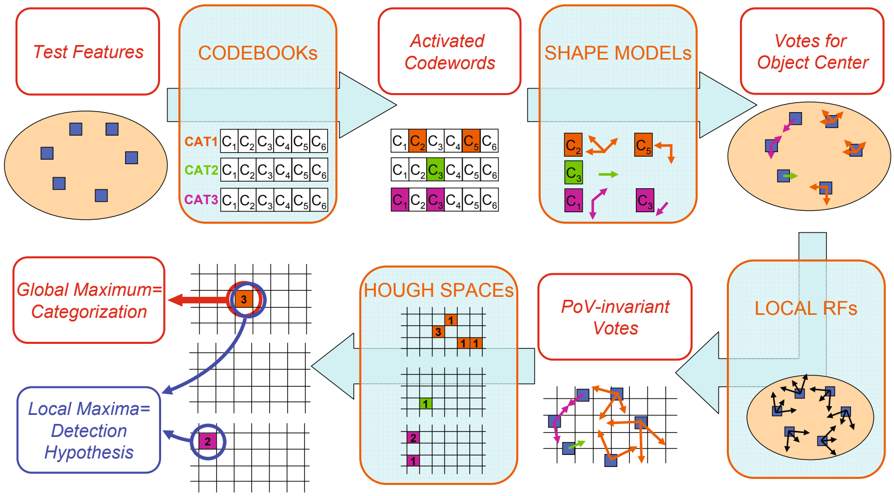


Fig. 2. Overview of 3D ISM for Categorization and Detection

repeatable direction on the tangent plane to define a full 3D local RF. However, SHOT [13] is a recent 3D descriptor proposal that includes a repeatable local RF and yields state-of-the-art performance. We thus use these features throughout this work. In turn, one of the contribution of this paper is to show that such recently proposed features demonstrate good performance even in 3D object categorization, an experiment that was not proposed in [13].

In the previous overview of the method we have highlighted the main design decisions that need to be taken to define a 3D ISM, i.e. the *codeword activation strategy* and the *vote weighting strategy*. In the following we address, by discussion and experiments, the possible alternatives for these design choices together with other major issues related to *codebook size and composition*. It is worth noting that, although we have conducted experiments using 3D data only, all our reasoning is independent from data dimensionality. Therefore, we expect the observations drawn from our analysis to be beneficial also for the case of standard 2D ISMs.

3 Codebook

3.1 Codebook Size

Codebooks are widely used for 2D and 3D object categorization (e.g. [14] [10] [11]). The reason behind their use is efficiency, both in terms of memory occupancy of the codebook and computational time for codeword activation. They are not expected to have any positive impact on the generalization abilities of the algorithms. They are usually built by applying some standard clustering algorithms, like K-Means, on the features extracted from the training data. Little attention, however, has been paid to the loss in discriminative power of the

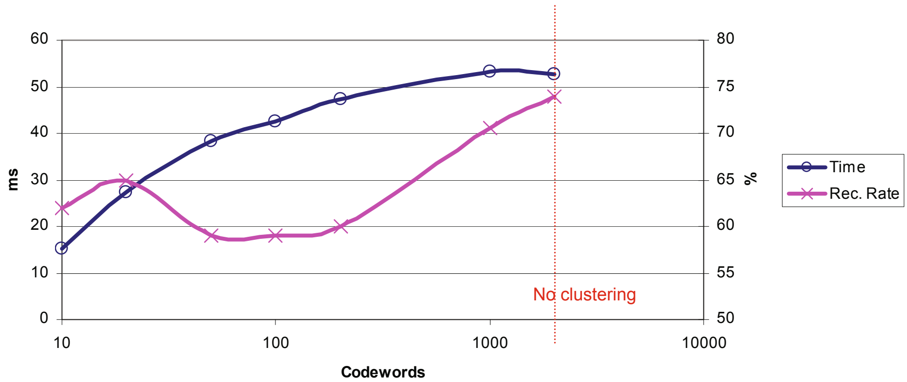


Fig. 3. Impact of codebook size on mean recognition rate and mean recognition time

codebook after size reduction. Furthermore, research in the field of Approximate Nearest Neighbor provides us efficient methods to solve the codeword activation problem even in high dimensional spaces and with large databases [15]. Finally, the cost of storing a set of descriptors for each training model of the currently publicly available 3D datasets is nowadays definitely affordable by off-the-shelf machines. Based on the above considerations, we investigated on the actual importance of building a codebook to successfully perform object category recognition in 3D data.

The chart in Fig. 3 shows the outcome of an experiment carried out on the Aim@Shape Watertight dataset (see Sec. 6 for more details about the dataset and the experimental methodology). We used half dataset for training and half for testing, i.e. ten models for training and ten for testing for each category. 200 mesh vertexes were randomly selected on each training model obtaining 2000 features as training set for each category. We then performed K-Means on this set, varying K logarithmically from 10 to 2000. We used such codebooks to categorize the test set. The best mean recognition rate is obtained with 2000 codewords, i.e. using the plain training data without any clustering. Loss in efficiency is minimal, for instance using 100 codewords the mean time to categorize one test model is about 42 ms, whereas using the plain training set as codebook it slightly increases to about 52 ms. Memory occupancy, of course, scales linearly with codebook size and, for the considered dataset, when using no clustering is less than 57MB. Therefore, based on the indication of this and other similar experiments, in the following we use as "codebook" the whole training data, without carrying out any clustering on them.

3.2 Sharing Codewords among Categories

In the original ISM proposal, the case of simultaneous recognition of multiple categories is solved by running a detector for each category, endowed with its own codebook built from training data belonging to its category. We refer to this configuration as ISM with *separated codebooks*: codebooks of different categories are independently built and used. In the context of categorization of DB entries, we have investigated on another possible configuration, that we refer to here as ISM with *global codebook*: a codebook is created from the training data belonging to all categories and then used by all ISMs. The Shape Model of each category is still built during the training stage by considering only the training data belonging to that category. However, denoting with SM_i the Shape Model of category C_i, not only those originated by the training data of C_i, but all the codewords in the codebook, regardless of the categories of the features that generated them, can participate to SM_i, provided that they are similar - according to the codeword activation strategy - to any of the training features of C_i. Therefore, this scheme endows the ISM paradigm with a broader capability of generalization: whilst the separated codebooks configuration is able to generalize at an intra-class level, by letting features observed in different training instances of the same class collaborate to the detection of an instance during testing, the global codebook configuration lets ISM generalize also at an *inter-class* level. It allows features observed in training examples of different categories to reinforce the hypothesis that an instance of category C_i is present. In other words, it builds a "universal" codebook of all the likely features given the training data, and then associates a spatial location for a specific category to all those that are "similar" to the training features of such category, regardless of the labels of the training data that originated that codeword.

It is worth highlighting that memory requirements of both configurations are equal: although a global codebook requires C times more space than a separated codebook, with C the number of categories, only one instance of it has to be stored in memory since it can be shared among all the C 3D ISM required by our proposal. Query time scales logarithmically with the size of the codebook: since codewords in the global codebook are C times those of the separated codebooks, query time is increased by $\log C$, a limited amount for typical number of categories in publicly available 3D databases (i.e. less than 30).

4 Codeword Activation Strategy

The codeword activation strategy proposed for the deployment of ISM in the case of 2D data [5] is the *cutoff threshold*: codewords are activated, and, thus, cast their votes, if their distance from the test feature is below a threshold. An alternative approach is represented by the k-NN activation strategy: the closest k codewords to the test feature are activated, regardless of their distance. We consider the latter strategy more suitable to the task of categorization, the reason being twofold. First of all, in those parts of the feature space characterised by a high codeword density, k-NN activates generally less features than the cutoff strategy, only the k most similar ones. By increasing the number of votes casted by each test feature in the Hough Space we may expect to sharpen the peak corresponding to a true instance of the class, but also to generate spurious peaks in the voting space, by randomly accumulating wrong votes in the same bin. In such parts of the feature space, the k-NN strategy acts as a filter that aims at reducing the probability of adding noise into the Hough Space, while it hopefully retains the ability to let the correct hypothesis emerge, by selecting only the most similar codewords. Secondly, in those parts of the feature space with a low density or even absence of codewords, k-NN activates anyhow k codewords, whereas the cutoff strategy cast very few votes, if any. Indeed, being the threshold generally chosen as small as to prevent generation of false peaks, the cutoff strategy generally tends not to activate any codeword in low density regions of the feature space. Obviously, the codewords activated by the k-NN strategy can be really different from the test data. Still, given the training set, they are the most similar available: if we have to generalize from the training examples to attempt to classify the current input, they appear a reasonable choice. The same reasoning does not hold when using 3D ISM to detect instances in cluttered scenes: in such a case, a high distance from any codeword is likely to indicate that the test feature comes from clutter and hence should not cast votes, such behavior being correctly modeled by the cutoff strategy. Yet, when reasoning in absence of clutter, as it is the case of categorization of entries of a 3D database, the k-NN strategy offers an adaptive behavior with respect to the training data that seems more suitable to the task.

5 Votes Weighting Strategy

In [5], the vote weight for each pair (test feature, vector in the shape model) is given by the product of a match weight and an occurrence weight

$$w = p\left(o_n, x|C_i, l\right) * p\left(C_i^*|f_k\right) = \frac{1}{|M|} * \frac{1}{|Occ\left[i\right]|} \tag{1}$$

with M being the set of codewords activated by the test feature f_k and $Occ[i]$ being the set of vectors in the Shape Model associated with codeword i.

The rationale behind this choice is tightly coupled with the use of the original ISM for detection in cluttered scenes. In presence of clutter, there is an obvious trade off between increasing the number of true detections and limiting the number of false detections. The choice of the vote weighting strategy operated in [5] goes in this direction. If a feature activates more codewords than another feature and/or if such codewords can be observed in more feasible positions with respect to the object center than other codewords, then this feature will be regarded as less distinctive since it likely generates more spurious votes in the Hough Space. By keeping low the weight, i.e. the confidence, on the position of the object center for the votes of such features, the original ISM tries to choose a good working point to optimize the above mentioned trade-off, by keeping below the detection threshold such spurious local maxima of the voting space. We refer to this vote weighting strategy as *Localization Weights (LW)*.

Again, in absence of clutter the scenario is different. Recall from Sec. 2 that we propose to select as output the category yielding the global maximum among all the Hough spaces. Therefore, in this case the emphasis for each 3D ISM should be on supporting as much as possible its best hypothesis. This means that spurious local maxima are not relevant for categorization, as long as they do not hide the true global maximum. Since we can reasonably expect that the geometrical consistent bin will likely be the strongest peak in the voting space, there is no reason to try to weaken local maxima by acting on the vote weight. On the other hand, using the original ISM vote weighting strategy may uselessly reduce the strength of the global maximum only because features that casted vote for it have also casted votes for wrong locations, and this can lead to a wrong selection of the correct category in the final competition among each global maximum of all categories. Hence, in the case of categorization, we have investigated on the use of the same constant weight for all features and codewords. Hereinafter, we will denote this vote weighting strategy as *Categorization Weights (CW).*

6 Experimental Results

We have tested our proposals on the Aim@Shape Watertight (ASW) dataset, previously used for the evaluation of 3D object categorization algorithms such as [10], and on the Princeton Shape Benchmark (PSB) [16], already used for 3D categorization in [11]. Since meshes in the PSB dataset exhibit a high variance in metric dimensions, even within the same class, to define a Hough Space suitable for all meshes, we normalize models before using them for testing or training. Specifically, we translate the model barycenter into the origin, compute the Eigenvalue Decomposition (EVD) of the scatter matrix of each model to find its principal axes, we scale the model down or up by a scale factor given by $1/X_{max} - X_{min}$, with X_{max},X_{min} the maximum and minimum coordinates of the mesh along the first principal axis, and finally rotate the model to align it with its principal axes. It is important to note that, due to the sign ambiguity inherent to the EVD (see e.g. [13]), we still need PoV-independent votes to achieve correct categorization. This normalization allows also for an important simplification: we can define the Hough Space just around the barycenter, i.e. the origin: any hypothesis for the object center laying far away from the barycenter will clearly be a spurious peak in the voting space. This improves both the effectiveness and the efficiency of our method, since it reduces the memory footprint needed to store the Hough Space. In particular, we used a Hough Space consisting of one squared bin, centered in the origin and with a side of 0.2. In all the experiments with both datasets we randomly extract 200 feature points from each training model and 1000 feature points from each testing model, and we describe them using SHOT with 16 spatial sectors (8 on the tangent plane and 2 concentric spheres) and 10 bins for the normal histograms. We do not perform any multi scale description, we use just a single support radius, equal to 0.25 and 0.45 for the AWS and the PSB dataset, respectively. As discussed in section 3.1, we use a plain codebook composed by all training descriptors.

The Aim@Shape Watertight dataset contains 20 categories, each composed of 20 models. We tested our performance on this dataset according to two methodologies. First, we divided the dataset in a training and a testing set by taking the first 10 models of each category as training set and the rest as testing set. With

	Human	Cup	Glasses	Airplane	Ant	Chair	Octopus	Table	Teddy	Hand	Plier	Fish	Bird	Spring	Armadillo	Bust	Mech	Bearing	Vase	Fourleg
Human	80									10					10					
Cup		90													10					
Glasses			100																	
Airplane				100																
Ant					100															
Chair						100														
Octopus							30		20	20					20					10
Table								100												
Teddy									100											
Hand										90		10								
Plier											100									
Fish									10			80	10							
Bird	10			10			10						60		10					
Spring				10	10					10	20			40	10					
Armadillo															100					
Bust									20							70			10	
Mech																	100			
Bearing								20			10							70		
Vase											10				10	40		10	30	
Fourleg												20			40					40

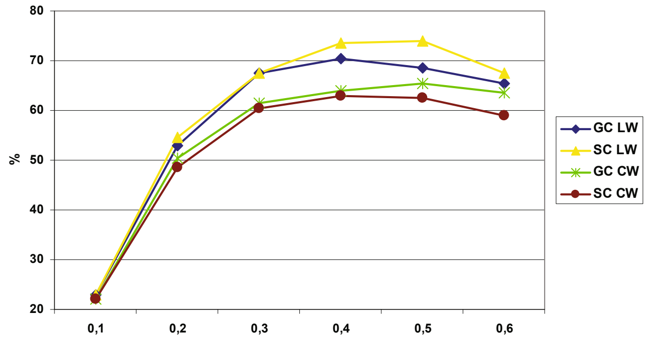
Fig. 4. Confusion Matrix for Aim@Shape Watertight, 1-NN Codeword Activation Strategy and CW Votes Weighting Strategy. The rows represent the test categories of the input model, the columns the output of the 3D ISM.

this configuration we studied the influence of the previously discussed design issues. Then, we also performed Leave-One-Out cross validation as done in [10], to be able to compare our results with such related work. Of course, the first test is more challenging, since significantly less training data is available to learn category shapes.

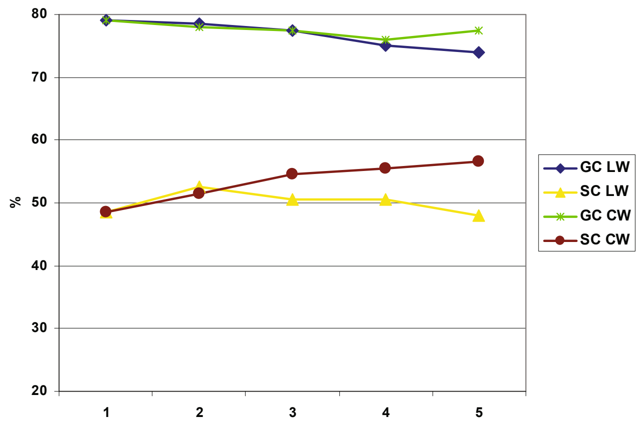
Results for the first series of experiments are reported in Fig. 5. We compared the performance of all the combinations of the proposed design decisions, i.e. global codebook (GC) vs. separated codebooks (SC), LW vs. CW and k-NN vs. cutoff with different values. The best recognition rate for this dataset is 79% and is obtained using 1-NN as Codeword Activation Strategy and a global codebook. In such configuration LW is the same as CW, since each codeword has zero or one vote. Fig. 4 reports the confusion matrix for such case.

In the case of the Leave-One-Out cross validation, [10] reports a mean recognition rate of 87.25%. Using 2-NN as Codeword Activation Strategy, a global codebook and CW as Votes Weighting Strategy, we have obtained **100**%.

The PSB dataset comes with a hierarchical categorization and a predefined division in training and testing sets. We use such categorization and such division. To compare our results against those in [11] we use the categorization level named Coarse 2, although it defines quite abstract meta-categories, such as "Household", which includes electric guitars, guns as well as stairs, or "-1", that stands for "all other models in the dataset". Clearly this dataset is more challenging than ASW, the intra-class and the inter-class variability being definitely higher.



(a) cutoff



(b) k-NN

Fig. 5. Mean recognition rate as a function of varying cutoff and k-NN values on Aim@Shape Watertight

Results are reported in Fig. 6. We compared the same combinations as in the previous experiment. The best recognition rate for this dataset is 50.2% and is obtained using 2-NN as Codeword Activation Strategy, a global codebook and the CW Votes Weighting Strategy. [11] reports a mean recognition rate of 55%. It is worth noting that, in addition to the previously mentioned difficulties, the PSB dataset presents also a highly variable point density among the models. As it has been noted in [13], point density variation is not well tolerated by current 3D descriptors. This was explicitly accounted for in [11], where all PSB meshes were resampled to a constant number of vertexes, uniformly distributed in the meshes. We have not implemented such resampling yet, that could likely improve our performance.

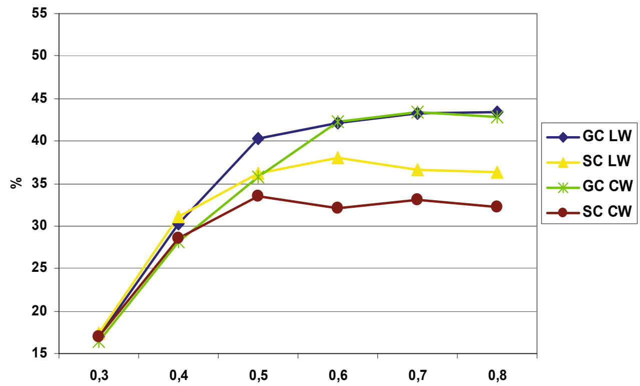
7 Discussion

The most evident outcome of our investigation is definitely the fact that the Codeword Activation Strategy and codebook composition play a significant role on the performance of 3D ISM for categorization. In both datasets k-NN with global codebook consistently outperforms the cutoff threshold with both kinds of codebook composition, regardless of the choice of k. This confirm two intuitions: a) that the intrinsic adaptation to codewords density in the feature space provided by k-NN is more suitable for database entries categorization, i.e. in absence of clutter, since it enhances ISM generalization ability; b) that the global codebook, when compatible with the application constraints on memory occupancy and computation time, endows ISM with higher, inter-class generalization power.

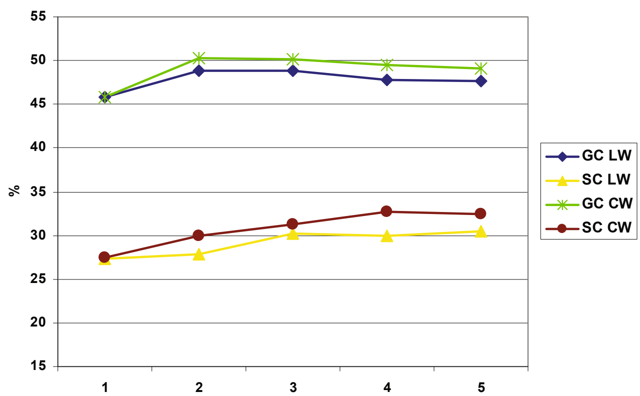
Experiments also reveal a tight coupling between the use of k-NN and the global codebook: k-NN with separated codebooks exhibits unsatisfactory performance, even with respect to the cutoff strategy. With the global codebook the k nearest neighbor codewords for a test feature are the same for each tested category, i.e. they represent the overall k most similar features throughout those belonging to all categories seen in the training stage, what then differs for the different categories is how these codewords vote in the different ISMs. In particular, it is worth pointing out that, differently from the case of separated codebooks, it happens that some of the codewords have no associated votes in the ISM of a specific category. This happens when a codeword is not similar to any training data of that category. Therefore, many of the k activated codewords will likely vote only for a subset of the categories, so that votes accumulation in the Hough Space has more chances to let the true category emerge, being required to filter out a limited amount of wrong votes. In other words, this configuration balances the impact of codebook (i.e. of features similarity) and shape model (i.e. of geometrical structure) and results in good recognition rates. With separated codebooks, instead, the k nearest neighbors are different in different codebooks, so that in several of them the activated codewords may be very dissimilar to the test feature. Moreover, since there are no codewords without votes in this configuration, all the activated codewords will cast votes in their shape models. This configuration, therefore, tends to diminish the importance of feature similarity and relies almost completely on shape models being able to select the correct category. This increases the probability of generating wrong, spurious peaks in the voting space.

The vote weighting strategy does not play a role as important as the other two discussed design decisions. Nevertheless, as far as the k-NN codeword activation strategy is concerned, the Categorization Voting obtains consistently slightly better performance in both datasets and with both kind of codebooks. This provides experimental evidence to the reasoning of Sec. 5.

As for the experiments on the cutoff threshold strategy, whilst on the PSB dataset the global codebook is still the favorable option, and there is little difference between the votes weighting strategies, in the case of the ASW dataset the decisive factor for obtaining higher performance seems to be the LW strategy whereas, unlike in the k-NN case, the codebook options seem to have quite a minor impact. We ascribe the latter to the cutoff strategy intrisecally balancing feature similarity and geometrical structure, for dissimilar codewords, given the cutoff threshold, cannot cast votes at recognition time also when the separated



(a) cutoff



(b) k-NN

Fig. 6. Mean recognition rate as a function of varying cutoff and k-NN values on the PSB coarse 2 dataset

codebook is used. On the other hand, it is quite more difficult to explain the higher performance of LW on this dataset. The higher performance of LW seems to suggest that in the ASW dataset wrong categories are supported in the voting space by less distinctive codewords, whose vote weights are indeed diminished by using LW.

The Confusion Matrix in Fig. 4 evidences how, beside gross errors that must be accounted to the difficulty of the task, several errors are somehow reasonable for an algorithm that tries to categorize objects based only on 3D shape. For instance, the category "Octopus", for which our proposal fails to recognize the majority of test models, is confused with "Hand", "Armadillo" and "Fourleg", i.e. with categories that present sort of "limbs" in configurations similar to those assumed by the models in the "Octopus" category. The 40% of "Fourleg" test models are wrongly categorized as "Armadillo", which, again, in some training models appears in a Fourleg-like pose. All the wrongly assigned test models of

"Bearing" are labeled as "Table" or "Plier", which have parts (the legs, the handles) that are shaped as bearings. Provided that this dataset can be successfully categorized by using only shape when enough training data can be deployed, as our 100% result in the Leave-One-Out test demonstrates, the mostly reasonable errors in the Confusion Matrix show that our proposal is able to learn a plausible, although less specific, model for the category shape in presence of less training data.

8 Conclusions

We have presented a new proposal for categorization of 3D data, which relies on the deployment of Implicit Shape Models in combination with a recently proposed 3D descriptor. We have devised the general structure of a 3D ISM and, based on its analysis, identified and discussed three design decisions that could improve the performance of the method when used for categorization. Experimental results on two well known and relative large datasets demonstrate that the combination of the k-NN codeword activation strategy and the use of a global codebook built from the whole training data of all categories is more suited to categorization than the standard ISM approach. Votes weighting strategy, on the other hand, does not seem to play such an important role for overall performance. The proposed optimal configuration compares favorably with the state of the art in 3D data categorization, obtaining similar results in one case and outperforming current proposals on the other considered dataset.

References

1. Pinz, A.: Object categorization. Foundation and Trends in Computer Graphics and Vision 1, 255–353 (2005)
2. Csurka, G., Bray, C., Dance, C., Fan, L.: Visual categorization with bags of keypoints. In: ECCV Workshop on Stat. Learning in Computer Vision (2004)
3. Sivic, J., Russell, B., Elfros, A., Zisserman, Z.: Discovering objects and their location in images. In: Proc. ICCV (2005)
4. Serre, T., Wolf, L., Poggio, T.: A new biologically motivated framework for robust object recognition. In: Proc. CVPR (2005)
5. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an Implicit Shape Model. In: Proc. ECCV, pp. 17–32 (2004)
6. Tangelder, J.W.H., Veltkamp, R.C.: A survey of content based 3D shape retrieval methods. In: Proc. Shape Modeling International, pp. 145–156 (2004)
7. Iyer, M., Jayanti, S., Lou, K., Kalyanaraman, Y., Ramani, K.: Three dimensional shape searching: state-of-the-art review and future trends. Computer Aided Design 5, 509–530 (2005)
8. Johnson, A., Hebert, M.: Using spin images for efficient object recognition in cluttered 3D scenes. PAMI 21, 433–449 (1999)
9. Frome, A., Huber, D., Kolluri, R., Bülow, T., Malik, J.: Recognizing objects in range data using regional point descriptors. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3023, pp. 224–237. Springer, Heidelberg (2004)

10. Toldo, R., Castellani, U., Fusiello, A.: A *bag of words* approach for 3D object categorization. In: Gagalowicz, A., Philips, W. (eds.) MIRAGE 2009. LNCS, vol. 5496, pp. 116–127. Springer, Heidelberg (2009)
11. Liu, Y., Zha, H., Qin, H.: Shape Topics: a compact representation and new algorithms for 3D partial shape retrieval. In: Proc. CVPR (2006)
12. Ohbuchi, R., Osada, K., Furuya, T., Banno, T.: Salient local visual features for shape-based 3D model retrieval. In: Proc. Int. Conf. on Shape Modeling and Applications, pp. 93–102 (2008)
13. Tombari, F., Salti, S., Di Stefano, L.: Unique signatures of histograms for local surface description. In: Proc. European Conference on Computer vision (ECCV 2010) (2010)
14. Sivic, J., Zisserman, A.: Video google: Efficient visual search of videos. In: Ponce, J., Hebert, M., Schmid, C., Zisserman, A. (eds.) Toward Category-Level Object Recognition. LNCS, vol. 4170, pp. 127–144. Springer, Heidelberg (2006)
15. Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. In: International Conference on Computer Vision Theory and Application VISSAPP 2009, pp. 331–340. INSTICC Press (2009)
16. Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The Princeton Shape Benchmark. In: Shape Modeling International (2004)