Smooth Object Retrieval using a Bag of Boundaries

Relja Arandjelović Andrew Zisserman
Department of Engineering Science
University of Oxford
{relja,az}@robots.ox.ac.uk

Abstract

We describe a scalable approach to 3D smooth object retrieval which searches for and localizes all the occurrences of a user outlined object in a dataset of images in real time. The approach is illustrated on sculptures.

A smooth object is represented by its material appearance (sufficient for foreground/background segmentation) and imaged shape (using a set of semi-local boundary descriptors). The descriptors are tolerant to scale changes, segmentation failures, and limited viewpoint changes. Furthermore, we show that the descriptors may be vector quantized (into a bag-of-boundaries) giving a representation that is suited to the standard visual word architectures for immediate retrieval of specific objects.

We introduce a new dataset of 6K images containing sculptures by Moore and Rodin, and annotated with ground truth for the occurrence of twenty 3D sculptures. It is demonstrated that recognition can proceed successfully despite changes in viewpoint, illumination and partial occlusion, and also that instances of the same shape can be retrieved even though they may be made of different materials.

1. Introduction

Recognizing specific objects, such as buildings, paintings, CD covers etc is to some extent a solved problem – provided that they have a light coating of texture. This success is a result of extensive research into viewpoint and lighting invariant feature detection [14, 18], feature description [14, 17], and the introduction of scalable methods based on bags of visual words [25]. There have been several large scale demonstrations [11, 21, 22] with Google Goggles as a commercial application. The focus of research in this area has largely moved on from feature detection development, to learning feature descriptors [24, 27] and overcoming various failings of the recognition pipeline such as the vector quantization into visual words and the problem of regular patterns [7, 8, 11, 12, 16, 19, 23].

However, as has been noted for quite some time [18, 20], there are two classes of specific objects for which current methods fail completely: wiry objects [6] and smooth (fairly textureless) objects. This paper addresses the smooth object class.

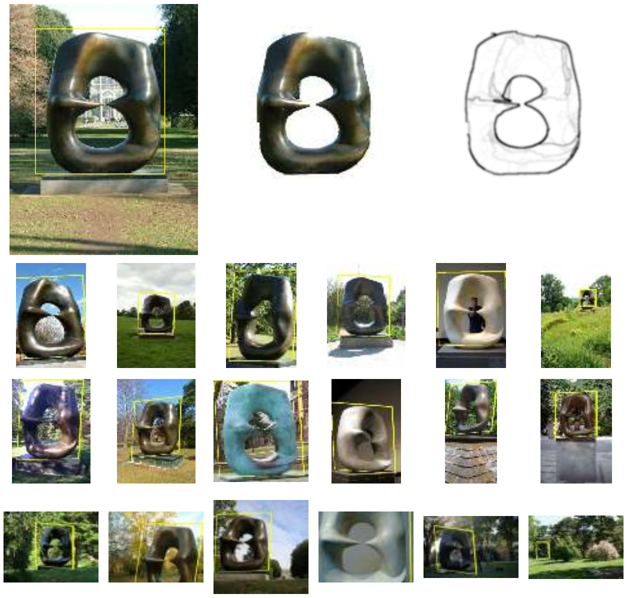


Figure 1. **Smooth sculpture retrieval using a bag of boundaries (BoB).** Top row: (left) a sculpture by Henry Moore selected by a user-outlined query; (middle) automatically segmented sculpture (section 2.1); (right) the boundary and internal edges are represented using semi-local descriptors (section 2.2) and indexed using a BoB (section 3). Bottom three rows: 18 of the retrieved images in rank order (before any false positives) showing the BoB's robustness to scale, viewpoint, lighting, colour and material variations. Note, at least seven different instances of the sculpture are retrieved, made out of at least three different materials.

Our goal is to raise smooth objects to the first class status that lightly textured specific objects have: to be able to recognize these objects under change of lighting; and under change of scale and viewpoint; and to be able to build scalable retrieval systems. In this work we consider smooth objects that are three dimensional (3D), and will use sculptures as our illustration. We are interested in matching objects of the same shape, and for sculptures, where the same form may be produced multiple times, this means that two instances may have the same shape but differ in size and even material. For example, Henry Moore routinely made the same sculptural form in bronze and marble.

3D smooth objects also bring with them the additional issue that their boundaries (internal and external) depend on viewpoint since they are defined by tangency with the line of sight [13]. This means that the imaged shape can vary continuously with viewpoint, and we address this for the moment by a view based representation.

To this end we develop a new representation for smooth objects that encodes their boundaries (internal and external) both locally and at multiple scales (section 2.2). This representation is inspired by the shape context descriptors of Belongie and Malik [5] and also by the silhouette representation used by Agarwal and Triggs [3]. We show that this representation is suitable for matching smooth 3D objects over scale changes, and is tolerant to viewpoint change and segmentation failures.

However, the representation cannot be employed directly for objects in an image due to the overwhelming number of edges and boundaries in the background (from clutter, trees, people etc). Instead it is first necessary to improve the signal to noise (where signal is the sculpture) by segmenting the image to isolate the sculpture as foreground. We show that this can be accomplished quite successfully using a combination of unsupervised segmentation into regions [4] and supervised classification of the regions [10] (section 2.1).

Finally, section 3, we show that the boundary representation can be vector quantized into a form suitable for large scale retrieval in a manner analogous to visual words [25]. As in the case of a bag of visual words, a bag-of-boundaries (BoB) can be used to retrieve a short list of images containing the object via an inverted index, and the images can then be re-ranked on spatial consistency in the manner of [22].

Since there are no existing datasets with ground truth for sculptures we introduce (in section 4) a new dataset – sculptures 6K – with ground truth annotation for twenty sculptures and their viewpoints, and consisting of works principally by Moore and Rodin. Figure 2 shows a random sample of the images. The dataset specifies a training/test split and this is used to assess the performance of the retrieval system. We compare to a number of baselines using conventional visual words based on viewpoint invariant detectors and descriptors (section 5).

2. Sculpture representation

In this section we describe the representation of the object boundary by a set of semi-local descriptors. In order to obtain this representation from a (cluttered) image it is first necessary to partition the image into sculpture and non-sculpture regions. This segmentation has two benefits: it improves the signal to noise, and also it provides an approximate scale for the descriptor computation. We begin with the segmentation, and then develop the boundary representation in section 2.2.

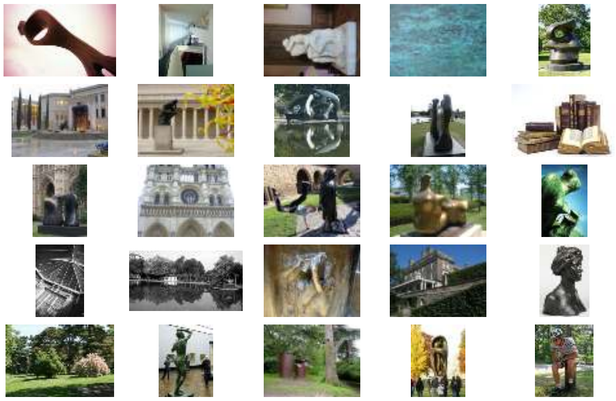


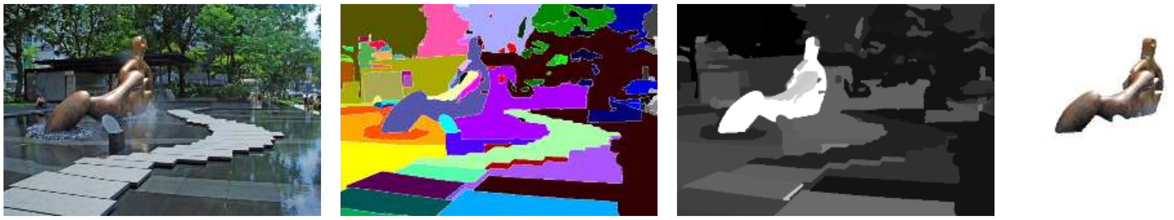
Figure 2. **Random samples from the Sculptures test dataset.** Note the variety of sculptures. Many of the images do not contain a sculpture while some contain people imitating the pose of a sculpture (*e.g.* bottom right image where a man is impersonating Rodin's *Thinker*).

2.1. Segmentation

The goal of the segmentation is to separate sculptures as foreground from the background. This is quite a challenging task since sculptures can be made from various materials including bronze, marble and other stone, and plastics. Their surface can be natural or finished in some way such as polishing (for stone) or buffered (for bronze) or even a light texture (*e.g.* deliberate chisel textures). The colour can include white, brown, specular highlights (on bronze), and even green (for algae or moss on outdoor installations). These must be distinguished from backgrounds that can have quite similar appearances including textureless sky, pavements and walls.

To achieve this segmentation we employ a supervised classification approach, engineering a feature vector that represents the appearance, shape and position of sculptures (relative to the image boundary). The segmentation proceeds in three steps: first, an over-segmentation of the image into regions (super-pixels); second, each super-pixel is classified into foreground (sculpture) or background to give an initial segmentation; and third, post-processing is used to filter out small connected components and obtain the final segmentation. Figure 3 illustrates these steps. Note, we do not attempt to group the super-pixels but simply classify them independently. We now describe these steps in more detail.

1. Super-pixels. We use the method and code from [4] which generates a hierarchy of regions based on the output of the gPb contour detector [15] This provides a partition of the image into a set of closed regions for any threshold. We use a threshold of 50 (out of 255) which yields about 58 super-pixels per image on average. A typical example of the super-pixels is shown in figure 3(b).



(a) Original image (b) Over-segmentation [4] (c) Classifier output, scaled for display (d) Final segmentation

Figure 3. **Automatic sculpture segmentation.** The image is in the Sculptures 6K test set.

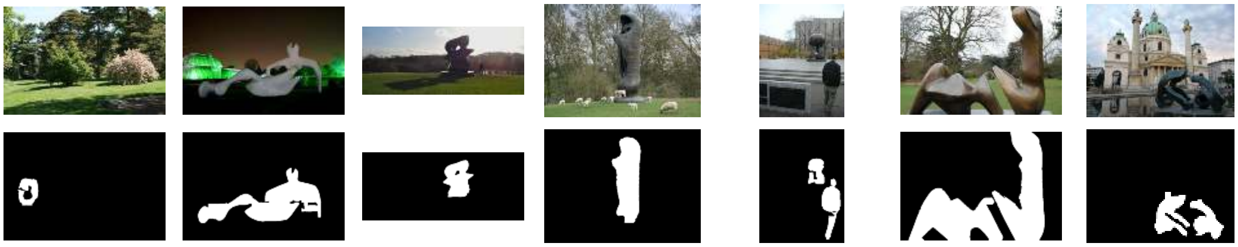
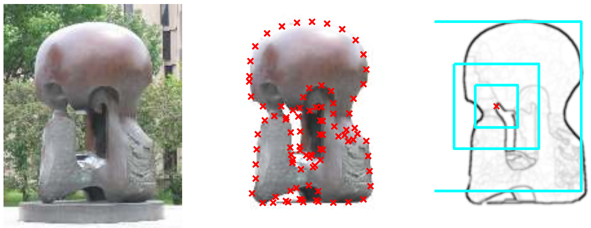


Figure 4. **Examples of automatic sculpture segmentation.** Top row shows images from the Sculptures 6K test set, bottom row shows the fully automatic segmentation.



(a) (b) (c)

Figure 5. **Boundary descriptor extraction.** (a) original image; (b) automatically segmented image (section 2.1) overlaid with the centres for the boundary descriptors; (c) boundary image with three different scaled descriptors centred at the same point. See section 2.2 for details.

2. Classification. For training and testing of the super-pixel classifier, 300 random images are selected from the Sculptures 6K training set, and segmented into super-pixels. The images are divided randomly into a training and validation set, each containing 150 images. Each super-pixel is then manually labelled into one of three classes: contained within a sculpture (positive example), not containing any sculpture pixels (negative example), or containing both sculpture and non-sculpture pixels (ignored completely). Small segments (less than 50×50 pixels) are also ignored in order to emphasize the correct classification of large segments.

Each super-pixel is described by a 3208 dimensional feature vector. This represents the appearance (colour, texture), shape and position of the segment (see below). A linear SVM classifier is trained on the annotated super-pixels from the training images, and its performance measured on the validation images. The histogram parts of the feature vector are compared using a χ^2 kernel, but using the efficient linear approximation of [26] enables a linear SVM to be used for these as well. The linear SVM leads to both fast training and testing.

The feature vector consists of: (i) the median gradient magnitude – this feature is typically very informative as its value is usually small for smooth object segments; (ii) four binary features indicating whether the segment is touching one of the image boundaries – in order to more easily distinguish sky, ceiling, wall and floor from smooth sculptures; (iii) colour represented by vector-quantized (using k-means, dictionary size 1600) HSV, and the mean HSV of the segment – this helps to identify the materials that sculptures are made of; and (iv) a bag of SIFT [14] visual words computed densely at multiple scales (dictionary size 1600, image patches with sides of 16, 24, 32 and 40, spacing of 2 pixels) – used for texture description, and also useful to identify sculpture material.

The super-pixel classifier has an accuracy of 96% on the training images, and 87% on the validation images. This results in a segmentation overlap score (intersection over union) of 0.78 on the training and 0.70 on the validation images.

3. Post-processing. The positive super-pixels are grouped using connected components, and small connected components (less than 50×50 pixels) of the foreground are removed. This does not significantly change the mean overlap score, but it removes many 'floating' and erroneous segments.

Examples of automatically segmented images are given in figure 4. These results show quantitatively and qualitatively that the automatic segmentation succeeds in its main objective of significantly increasing the signal (sculpture) to noise (other clutter) ratio.

2.2. Boundary descriptor

We develop a new shape descriptor suited for smooth object representation. Constructing such a descriptor is a challenging task as it needs to represent shape rather than texture or colour, be robust enough to handle lighting, scale and viewpoint changes, but simultaneously discriminative enough to enable object recognition. Additionally it should be extracted locally in order to be robust to occlusions and segmentation failures.

For an object, two types of descriptors are computed by sampling the object boundaries (internal and external) at regular intervals in the manner of [5]. They are (i) a HoG [9] descriptor, and (ii) a foreground mask occupancy grid. The scale of the descriptor is determined from the scale of the object. In order to represent the boundary information locally (*e.g*. the curvature, junctions) and also the boundary context (*e.g*. the position and orientation of boundaries on the other side of the object), the descriptors are computed at multiple scales. We use HoG computed on the gPb image here (rather than shape-context or SIFT for example) as we wish to represent both the position and orientation of the boundaries, and also their magnitude. Figure 5 illustrates the sampling and scales used.

In order to extract this representation from an image, it is first segmented into foreground (sculpture) and background as described above in section 2.1; and then the descriptor centres are obtained by sampling prominent foreground object boundaries and internal gPb edges at uniform intervals. The multiple scales of the descriptor are computed relative to the size of the foreground segmentation.

Implementation details. The first part of the descriptor uses 4×4 HoG cells, each containing 8×8 pixels (*i.e*. gPb patches are scaled to 32×32 pixels for HoG computation) and contrast insensitive spatial binning into 9 orientations, making the HoG part of the boundary description 324 dimensional (9 blocks each with 2×2 cells with 9 orientations). The HoG descriptor is L2 normalized in order to be able to compute similarities using Euclidean distance.

The second part of the descriptor is a 4×4 occupancy grid, where the value of a cell represents the proportion of pixels belonging to the foreground. The Hellinger kernel is used to compute similarities between these descriptors, *i.e*. the 16 dimensional descriptor is L1 normalized followed by square rooting each element thus producing a L2 normalized vector; the similarities are then computed using Euclidean distance.

The two parts of the descriptor are simply concatenated together making a 340 dimensional L2 normalized vector (the L2 norm being equal to 2).

The descriptor centres are obtained by sampling prominent (stronger than 30 out of 255) foreground object boundaries at uniform intervals. The interval lengths are determined as the maximum of 10 pixels and $1/50$ of the foreground object perimeter. The descriptor scales are set to be 1, 4 and 16 times $1/10$ of the foreground object area. Note that even though the largest scale descriptor is 1.6 larger than the object it does not in general cover the entire object as it is often computed at the external boundary, objects are usually elongated or not convex. The number of extracted descriptors per image is 450 on average.

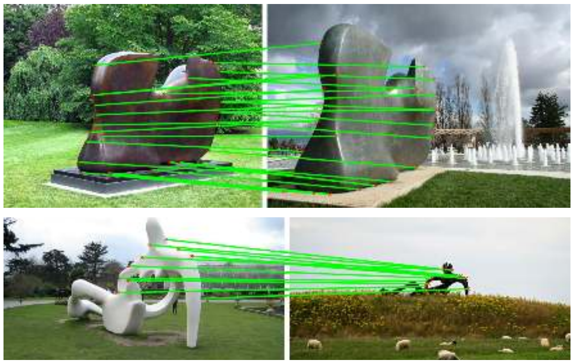


Figure 6. **Boundary descriptor matches.** Two examples of correctly matched sculptures using the semi-local boundary descriptor (section 2.2). Significant lighting, scale and viewpoint changes are handled well. Note that the images contain different sculpture instances but the shapes are identical and are successfully matched. Matches shown are after spatial verification (section 3).

Descriptor properties and matching. As the descriptor operates purely on boundary and segmentation data it is fairly unaffected by light, colour and texture changes. Scale invariance is obtained by computing the descriptor at multiple scales relative to the size of the foreground object they belong to. The descriptor is not rotation invariant but this can easily be alleviated by orienting the patch according to the boundary curve tangent.

The descriptors are matched between images using Euclidean distance. Note, even though three descriptors (at different scales) are computed at each of the sampled boundary points, these descriptors are matched independently in the subsequent processing – we do not explicitly enforce consistency between them. Figure 6 shows two examples of correctly matched sculptures, while figure 7 shows three typical retrieval results. Apart from illustrating robustness to lighting, colour, texture and scale differences, they also show that the descriptor is quite insensitive to significant viewpoint changes. There are three main reasons for this behaviour, firstly, the description is semi-local and even under significant viewpoint change it can be expected that some boundaries (and thus the descriptors) remain unchanged. Secondly, even though the object silhouette can change drastically between views, internal edges, which our method takes into account, can be unaffected. Finally, HoG cells inherently allow for some deformation in the position and orientation of boundaries.

The semi-local descriptors can be matched directly be-

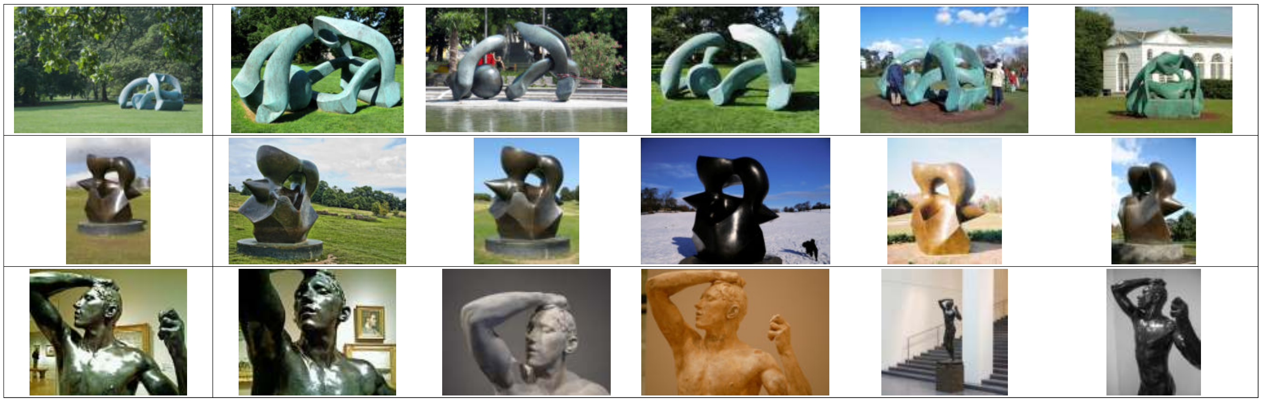


Figure 7. **Viewpoint invariance.** Each row shows one query (the left image), and the other five images are samples from the retrieved results. These results are typical, and demonstrate the viewpoint tolerance of the semi-local boundary representation.

tween object boundaries. However, in the following section we describe how this set of descriptors is represented as a histogram (by vector quantization and counting) in the manner of [3].

3. Retrieval procedure

Here we use the standard retrieval pipeline of Philbin *et al.* [22], but instead of representing the image as bag-of-visual-words (BoW) based on SIFT descriptors [14] computed at affine covariant regions [18], we develop a *bag-of-boundaries (BoB)* representation. For each image, boundary descriptors are extracted as described in section 2.2 and vector quantized using k-means; a histogram of these quantized descriptors (which we will also refer to as 'words') is then used to represent an image. Note, this is a bag representation as no information about the spatial position of the descriptors is recorded in the histogram. A query BoB is compared to other BoBs in the dataset using the standard tf-idf [22] measure. The tf-idf scores can be computed efficiently for each image in the database using an inverted index, which enables real-time retrieval in large databases.

As shown in [22] spatial verification and re-ranking of the top tf-idf retrieved results can be done efficiently and proves to be useful as it improves precision by ensuring spatial consistency between query and retrieved images. We adopt the same model for the geometry relation, namely an affine transformation. However, as the objects of interest are highly three-dimensional the affine model of the transformation is only approximate here, so only a very loose affine homography is fitted (*i.e.* large reprojection errors are tolerated) in order not to reject correct matches. We follow the procedure of [22] of first using a single (boundary) word match to determine a restricted affine transformation (in this case translation and scaling only), followed by fitting a full affine transformation to the inliers.

3.1. Implementation details

The BoB vocabulary is obtained from the Sculpture 6K test set descriptors. The test set generates 1.4M descriptors, we chose the vocabulary size to be 10k.

We spatially verify the top 200 results using a loose affine homography, tolerating reprojection errors of up to a 100 pixels. We also propose a new scoring system where the score of the geometrically verified image for a given query is computed as follows:

$$\text{score} = \text{tf-idf} + \alpha n + \beta \frac{n}{n_q}\frac{n}{n_r} \qquad (1)$$

where n_q and n_r are the number of words in the query and result images, respectively, and n is the number of verified matches. The proposed score is a generalization of the commonly used scoring scheme of [22] which corresponds to $\alpha = 1$ and $\beta = 0$. Our system accounts for the fact that images with many features are likely to have many spatially verified words and removes the bias from these images by considering the number of matches relative to the total number of features in the image.

4. Dataset and Evaluation

Sculptures 6K: We have collected a new image dataset in order to evaluate performance of smooth object retrieval methods. The dataset was obtained in a similar manner to the widely used Oxford Buildings dataset [22]: images containing sculptures were automatically downloaded from Flickr [1] using queries such as "Henry Moore Reclining Figure", "Henry Moore Kew Gardens" and "Rodin Thinker". The dataset has 6340 high resolution (1024×768) images.

The dataset is split equally into a train and test set, each containing 3170 images. For each set 10 different Henry Moore sculptures are chosen as query objects, and for each of these objects 7 images and query regions are defined, thus providing 70 queries for performance evaluation purposes. None of the 10 training set sculptures is present in the test set, whilst for the 10 test set sculptures mostly these are not present in the training set though there are a few occurrences as some images contain more than one sculpture (*e.g.* images taken in a museum). As well as the images containing these 10 sculptures in each set there are many images containing other sculptures or indeed no sculptures at all. These images act as distractors in retrieval. A sub-set of the test set queries is shown in figure 8.

For each query we have manually compiled the ground truth dividing all images into *Positives*, *Negatives* and *Ignores*: (i) *Negative* – No part of the queried sculpture is present. (ii) *Positive* – More than 25% of the queried sculpture surface is visible. (iii) *Ignore* – Less than 25% of the queried sculpture surface is visible, but the queried sculpture is present. Note that our definition of the *same sculpture* relationship requires two sculptures to have identical shapes, however it does not require them to be the *same instances* – they can be constructed of different materials, made in different sizes and displayed at different locations. Sculptures are 'highly' three-dimensional, unlike the building facades used in the Oxford Buildings dataset. For this reason the ground truth matches are *view specific* and vary over the different queries of the same sculpture. For example, it is unreasonable to expect to retrieve an image of a sculpture given an image taken from its opposite side. For each query the number of positive matches can vary from 5 to a maximum of 112, with a mean of 53.4.

The *Sculptures 6K* dataset with all the images and ground truth is available online at [2].

Performance evaluation: As in the case of the Oxford Buildings dataset, retrieval quality is evaluated using mean average precision (mAP) over all the queries. As in the INRIA Holidays [11] evaluation, the query image is not counted as a positive return (it is in the Oxford Buildings evaluation). In the mAP computation *Ignores* are not counted as positive or negative.

5. Results

To evaluate the performance of smooth object retrieval methods we follow the procedure outlined in section 4. The mean average precision (mAP) is computed over 70 queries on the test dataset.

Due to the lack of smooth object retrieval systems we use the standard affine-Hessian/visual word system of Philbin *et al*. [22] as a baseline (BL1). As a second baseline (BL2), we discard all visual words on the background (*i.e*. visual words are only included if their centres are in the automatically segmented foreground region). This is in order to give a fair comparison against our boundary representation which uses the foreground/background segmentation.

Retrieval performance. The mAP scores for the two baselines and our method are shown in table 1. As expected, there is a complete failure of the two baselines for smooth object retrieval. Note that BL2 perform slightly worse than BL1 (after spatial reranking with an affine homography) – this is due to the fact that many true positives in BL1 are actually obtained by matching the background of the sculpture installation instead of the actual queried sculptures. Note that none of the methods which usually improve retrieval performance can be hoped to help the two baselines: (i) query expansion [8] is only possible when the initial method yields high precision results which is certainly not the case here, (ii) soft vector quantization [11, 23], and (iii) learning a better vocabulary using [16, 19] both assume the descriptors to be appropriate for the task in hand which we demonstrate is not the case.

Method name	Spat. rerank	mAP	A.q.t.
Baseline 1 [22]		0.080	0.05 s
Baseline 1 [22]	√	0.094	0.30 s
Baseline 2 (Bg removed)		0.081	0.03 s
Baseline 2 (Bg removed)	√	0.086	0.11 s
BoB without seg.		0.253	0.01 s
BoB without seg.	√	0.323	0.16 s
BoB with segmentation		0.454	0.01 s
BoB with segmentation	√	**0.502**	0.28 s

Table 1. **Retrieval performance.** Comparison of two baseline bag of visual word methods (section 5) and the bag-of-boundaries (BoB) method (section 3). Mean average precision (mAP) scores and average query times (A.q.t.) are shown. The mAP scores correspond to the best choice of parameters (vocabulary size and reranking parameters α, β) for each method individually.

Our bag-of-boundaries (BoB) method proves to be very suitable for the task of smooth object retrieval, achieving more than a five fold increase in performance (0.502) over the best baseline (BL1, 0.094). The importance of the segmentation is shown by the 'with and without' comparison (*i.e*. in the 'with' case, only boundaries on the foreground region are used). There is 55% gain in performance for BoB when the foreground segmentation is used compared to using the entire image. On the other hand, the without segmentation performance is still quite respectable and demonstrates the robustness to background clutter. As would be expected, in some of the cases where automatic segmentation fails and the sculpture is assigned to background, the without case succeeds in retrieving the image. However, it is more prone to background clutter and less resistant to scale change as there is no scheme for automatic descriptor scale selection.

Examples of ranked retrieval results are given in figures 1, 7 and 9. They illustrate the appropriateness of the BoB system for the smooth object retrieval task as significant lighting, scale, viewpoint, colour and material differences are successfully handled.

Table 1 also gives the retrieval speed, tested on a laptop with a 2.67 GHz core i7 processor using only a single core. It can be seen that due to the inverted index implementation, the BoB representation enables real time retrieval to the same extent as visual words. The BoB representation is much sparser than the BoW (450 words per image for BoB compared to 2600 for BoW) making the entire storage re-

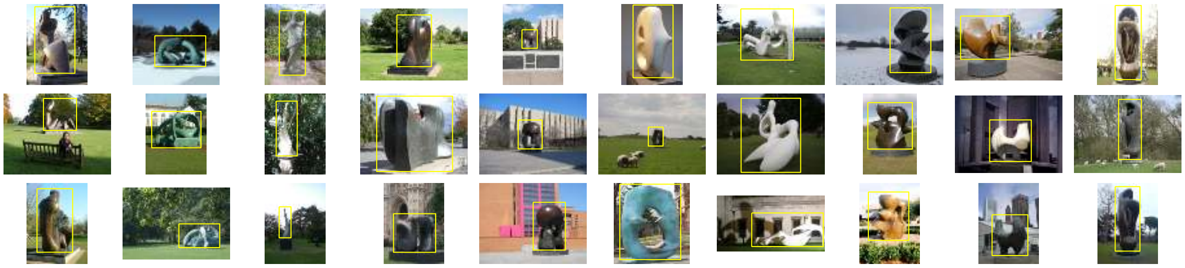


Figure 8. **Test dataset query images.** 30 query images (out of 70) used for evaluation in the Sculptures 6K test dataset. Each column shows 3 (out of 7) query images for one sculpture. Note the large variations in scale, viewpoint, lighting, material and background.

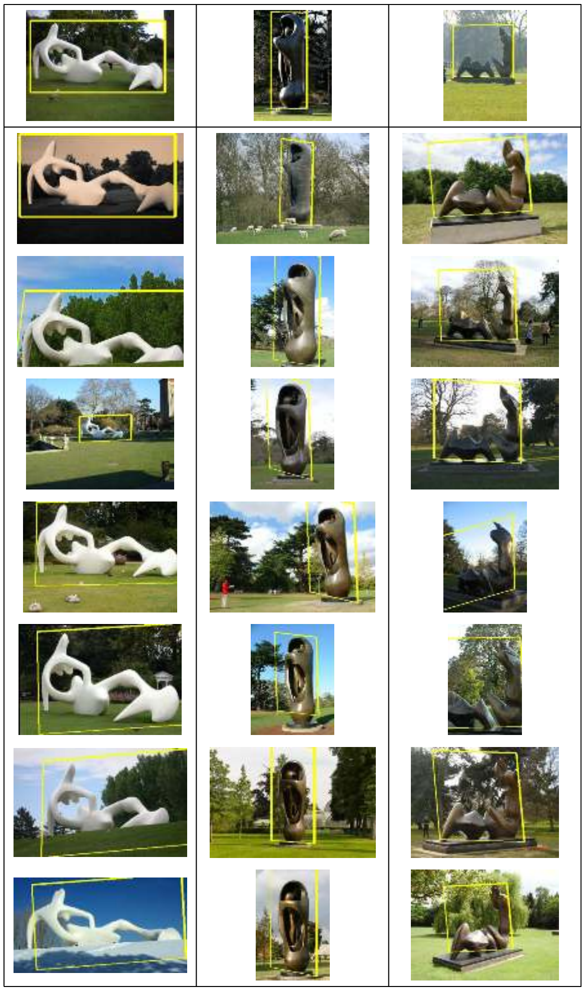


Figure 9. **Retrieval results.** Each column shows one retrieval result, the query image and ROI are shown in the first row, followed by the top 7 ranked retrieved images.

quirements for the system (inverted and forward indexes) a mere 20 MBs (in the BoW case this is 275 MBs). Our approach is thus much more scalable than the existing BoW ones as the BoB representation of up to 5 million images can fit into main memory on a system with 32 GB of RAM.

To further test the resistance to distractors, a larger scale retrieval test was performed by adding all 5062 images from the Oxford Buildings dataset to the testset. The mAP performance only dropped to 0.451 (from 0.502) despite the variety of images in the distractor dataset.

Due to the semi-local HoG boundary description (figure 5) and the BoB representation, the matching is capable of handling the significant segmentation failures that are bound to happen in a fully automatic system. The semi-local property means that a proportion of the HoG descriptors computed on the boundary will still be valid (the proportion depending on the extent of the segmentation failure), and the BoB representation enables matches for images where only a subset of the quantized descriptors are in common. Thus, as can be seen in figure 10, retrieval can succeed both in the cases of under-segmentation (where HoG descriptors will be missing) and over-segmentation (where additional erroneous boundaries are generated).

Parameter and descriptor variation. The choice of descriptor scales is critical for retrieval performance as reducing the areas by a factor of 4 reduces the mAP from 0.502 to 0.404. The problem with using small descriptors is that they are too local thus mainly capturing the orientation of a single edge, which without surrounding boundary information is completely non-discriminative.

Not using descriptors centred on internal boundaries but keeping the internal boundary information in the remaining descriptors reduces the mAP to 0.469, while not taking internal boundaries into account at all decreases it further to 0.433. This proves that using internal boundaries is very beneficial for shape representation.

The system is quite insensitive to the number of HoG and occupancy grid cells, using a coarser grid (3×3) decreases the mAP from 0.502 to 0.485 while using a finer one (6×6) increases it slightly to 0.509. The slight increase in performance when using a finer grid is not worth the large increase in descriptor dimensionality (from 340 to 936). Excluding the 16 dimensional occupancy grid part of the boundary descriptor decreases the mAP from 0.502 to 0.485, so it provides good value for the small descriptor dimensionality increase (from 324 to 340).

A main source of failure is due to the inevitable automatic segmentation mistakes, it is most prominent when the sculpture pixels in the query image are assigned to the back-

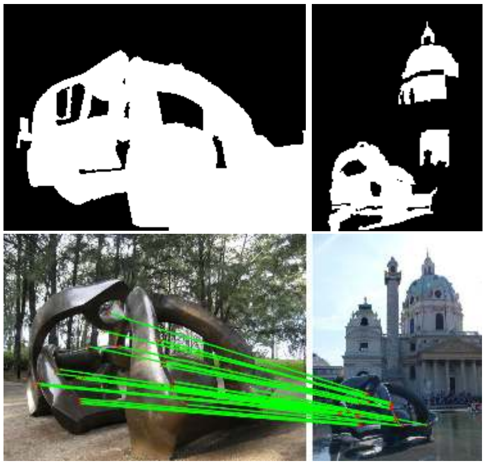


Figure 10. **Robustness to segmentation failures.** An example of a correctly matched sculpture despite significant segmentation failures. The segmentations are shown above the original images.

ground. This failure can be potentially alleviated by either segmenting the query image online or keeping descriptors for multiple alternative segmentations. The second failure mode is when all matched words occur spatially close to each other thus only effectively representing one part of the object which is not necessarily discriminative. A potential solution is to modify spatial verification to incorporate the information about the spatial distribution of matches, but this must be traded against robustness to occlusion and viewpoint change.

We investigate the effect of varying the reranking (section 3.1) parameter β for fixed $\alpha = 0$ on the mAP scores. The performance increases monotonically with β which effectively means the last portion of (1), which accounts for relative number of matched words, should dominate the reranking, while the tf-idf scores should be used for tie-breaking. Our reranking procedure, for the BoB method, always outperforms the reference reranking method [22] which uses just the unnormalized number of inliers and tf-idf. When a 10 times smaller vocabulary is used the benefits of the proposed reranking method are even more apparent (0.449 compared to 0.391) – words are less discriminative allowing the reference reranking method to incorrectly verify images with many features more easily thus reducing its precision.

6. Conclusions and further work

We have succeeded in our aim of raising 3D smooth objects to a first class specific object. This required both segmentation (a discriminative representation of the material appearance) and boundary representation. In doing this we have demonstrated that HoG can also be used as a descriptor for specific object retrieval (given suitably cleaned data), rather than solely as a descriptor where learning must be used. We have also established that 3D sculptures (as an example of 3D smooth objects) can be successfully retrieved using only a bag of boundary representation – without requiring any additional spatial information in the first instance.

We expect our framework to generalize to other classes of smooth objects, but new classifiers need to be trained to segment particular classes, *e.g.* plastic bottles, semi-transparent objects, etc.

Acknowledgements. We are grateful for financial support from the Royal Academy of Engineering, Microsoft, and ERC grant VisRec no. 228180.

References

[1] http://www.flickr.com/.
[2] http://www.robots.ox.ac.uk/∼vgg/research/sculptures/.
[3] A. Agarwal and B. Triggs. Recovering 3D human pose from monocular images. *IEEE PAMI*, 2006.
[4] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. From contours to regions: An empirical evaluation. In *Proc. CVPR*, 2009.
[5] S. Belongie and J. Malik. Shape matching and object recognition using shape contexts. *IEEE PAMI*, 2002.
[6] O. Carmichael and M. Hebert. Shape-based recognition of wiry objects. In *Proc. CVPR*, 2003.
[7] O. Chum and J. Matas. Unsupervised discovery of co-occurrence in sparse high dimensional data. In *Proc. CVPR*, 2010.
[8] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Proc. ICCV*, 2007.
[9] N. Dalal and B. Triggs. Histogram of Oriented Gradients for Human Detection. In *Proc. CVPR*, 2005.
[10] D. Hoiem, A. A. Efros, and M. Hebert. Geometric context from a single image. In *Proc. ICCV*, 2005.
[11] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Proc. ECCV*, 2008.
[12] H. Jégou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *Proc. CVPR*, Jun 2009.
[13] J. Koenderink. *Solid Shape*. MIT Press, 1990.
[14] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.
[15] M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik. Using contours to detect and localize junctions in natural images. In *Proc. CVPR*, 2008.
[16] A. Makadia. Feature tracking for wide-baseline image retrieval. In *Proc. ECCV*, 2010.
[17] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE PAMI*, 2004.
[18] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *IJCV*, 2005.
[19] A. Mikulik, M. Perdoch, O. Chum, and J. Matas. Learning a fine vocabulary. In *Proc. ECCV*, 2010.
[20] P. Moreels and P. Perona. Evaluation of features detectors and descriptors based on 3d objects. *IJCV*, 2006.
[21] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. CVPR*, 2006.
[22] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proc. CVPR*, 2007.
[23] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proc. CVPR*, 2008.
[24] J. Philbin, M. Isard, J. Sivic, and A. Zisserman. Descriptor learning for efficient retrieval. In *Proc. ECCV*, 2010.
[25] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, 2003.
[26] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. In *Proc. CVPR*, 2010.
[27] S. Winder, G. Hua, and M. Brown. Picking the best daisy. In *Proc. CVPR*, 2009.