Name that Sculpture

Relja Arandjelović
Department of Engineering Science
University of Oxford
relja@robots.ox.ac.uk

Andrew Zisserman
Department of Engineering Science
University of Oxford
az@robots.ox.ac.uk

ABSTRACT

We describe a retrieval based method for automatically determining the title and sculptor of an imaged sculpture. This is a useful problem to solve, but also quite challenging given the variety in both form and material that sculptures can take, and the similarity in both appearance and names that can occur.

Our approach is to first visually match the sculpture and then to name it by harnessing the meta-data provided by Flickr users. To this end we make the following three contributions: (i) we show that using two complementary visual retrieval methods (one based on visual words, the other on boundaries) improves both retrieval and precision performance; (ii) we show that a simple voting scheme on the tf-idf weighted meta-data can correctly hypothesize a subset of the sculpture name (provided that the meta-data has first been suitably cleaned up and normalized); and (iii) we show that Google image search can be used to query expand the name sub-set, and thereby correctly determine the full name of the sculpture.

The method is demonstrated on over 500 sculptors covering more than 2000 sculptures. We also quantitatively evaluate the system and demonstrate correct identification of the sculpture on over 60% of the queries.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.3.1 [**Information Storage and Retrieval**]: Content analysis and indexing; I.4.9 [**Image Processing and Computer Vision**]: Applications

General Terms

Algorithms, Experimentation, Performance

Keywords

Image retrieval, Object recognition, Image labelling

ICMR '12, June 5-8, Hong Kong, China

1. INTRODUCTION

The goal of this work is to automatically identify both the *sculptor* and the name of the *sculpture* given an image of the sculpture, for example from a mobile phone. This is a capability similar to that offered by Google Goggles, which can use a photo to identify certain classes of objects, and thereby carry out a text based web search.

Being able to identify a sculpture is an extremely useful functionality: often sculptures are not labelled in public places, or appear in other people's photos without labels, or appear in our own photos without labels (and we didn't label at the time we took them because we thought we would remember their names). Indeed there are occasionally pleas on the web of the form "Can anyone help name this sculpture?".

Identifying sculptures is also quite challenging. Although Google Goggles can visually identify objects such as landmarks and some artwork, sculptures have eluded it to date [11] because the visual search engine used for matching does not "see" smooth objects. This is because the first step in visual matching is to compute features such as interest points, and these are often completely absent on sculptures and so visual matching fails.

We divide the problem of identifying a sculpture from a query image into two stages: (i) visual matching to a large dataset of images of sculptures, and (ii) textual labelling given a set of matching images with annotations. Figure 1 shows an example. That we are able to match sculptures in images at all, for the first stage, is a result of combining two complementary visual recognition methods. First, a method for recognizing 3D smooth objects from their outlines in cluttered images. This has been applied to the visual matching of smooth (untextured) sculptures from Henry Moore and Rodin [4], and is reviewed in section 3.2. Second, we note that there is still a role for interest point based visual matching as some sculptures do have texture or can be identified from their surroundings (which are textured). Thus we also employ a classical visual word based visual recognition system. This is reviewed in section 3.1. The matching image set for the query image is obtained from the sets each of the two recognition systems returns (section 3.3).

The other ingredients required to complete the identification are a data set of images to match the query image to, and annotation (of the sculptor and sculpture name) for the images of this data set. For the annotated dataset we take advantage of the opportunity to harness the knowledge in social media sites such as Facebook and Flickr. As is well known, such sites can provide millions of images with some form of annotation in the form of tags and descriptions – though the annotation can often be noisy and unreliable [19]. The second stage of the identification combines this meta-information associated with the matched image set in order to propose the name of the sculptor and sculpture. The proposed sculpture name is finally determined using a form of query expansion from Google image search.

The stages of the identification system are illustrated in figure 1. We describe the dataset downloaded from Flickr in section 2, and the method of obtaining the name from the meta-data and Google query expansion in section 4.

Others have used community photo collections to identify objects in images [10, 12] and have dealt with the problems of noisy annotations [14, 21]. In particular, Gammeter *et al* [10] auto-annotated images with landmarks such as "Arc de Triomphe" and "Statue of Liberty" using a standard visual word matching engine. In [10], two additional ideas were used to resolve noisy annotations: first, the GPS of the image was used to filter results (both for the query and for the dataset); second, annotations were verified using Wikipedia as an Oracle. Although we could make use of GPS this has not turned out to be necessary as (i) sculptures are often sufficiently distinctive without it, and (ii) sculptures are sometimes moved to different locations (e.g. the human figures of Gormley's "Event Horizon" or Louise Bourgeois' "Maman") and so using GPS might harm recognition performance. Similarly, using Wikipedia to verify sculpture matches has not been found to be necessary, and also at the moment Wikipedia only covers a fraction of the sculptures that we consider.

2. DATASET

The dataset provides both the library of sculpture images and the associated meta-data for labelling the sculptor and sculpture. A list of prominent sculptors was obtained from Wikipedia [1] (as of 24th November 2011 this contained 616 names). This contains sculptors such as "Henry Moore", "Auguste Rodin", "Michelangelo", "Joan Miró", and "Jacob Epstein". Near duplicates were removed from the list automatically by checking if the Wikipedia page for a pair of sculptor names redirects to the same entry. Only Michelangelo was duplicated (as "Michelangelo" and "Michelangelo Buonarroti").

Flickr [2] was queried using this list, leading to 50128 mostly high resolution (1024 × 768) images. Figure 2 shows a random sample. For each of the images textual meta data is kept as well. It is obtained by downloading the title, description and tags assigned to the image by the Flickr user who uploaded it. The textual query (i.e. sculptor name) used to retrieve an image is saved too. This forms the *Sculptures 50K dataset* used in this work.

Unlike the recent Sculptures 6k dataset of [4] we did not bias our dataset towards smooth textureless sculptures.

3. PARTICULAR OBJECT LARGE SCALE RETRIEVAL SYSTEM

The first stage of the naming algorithm is to match the query image to those images in the *Sculptures 50k* that contain the same sculpture as the query. We briefly review here the two complementary visual retrieval engines that we have imple-

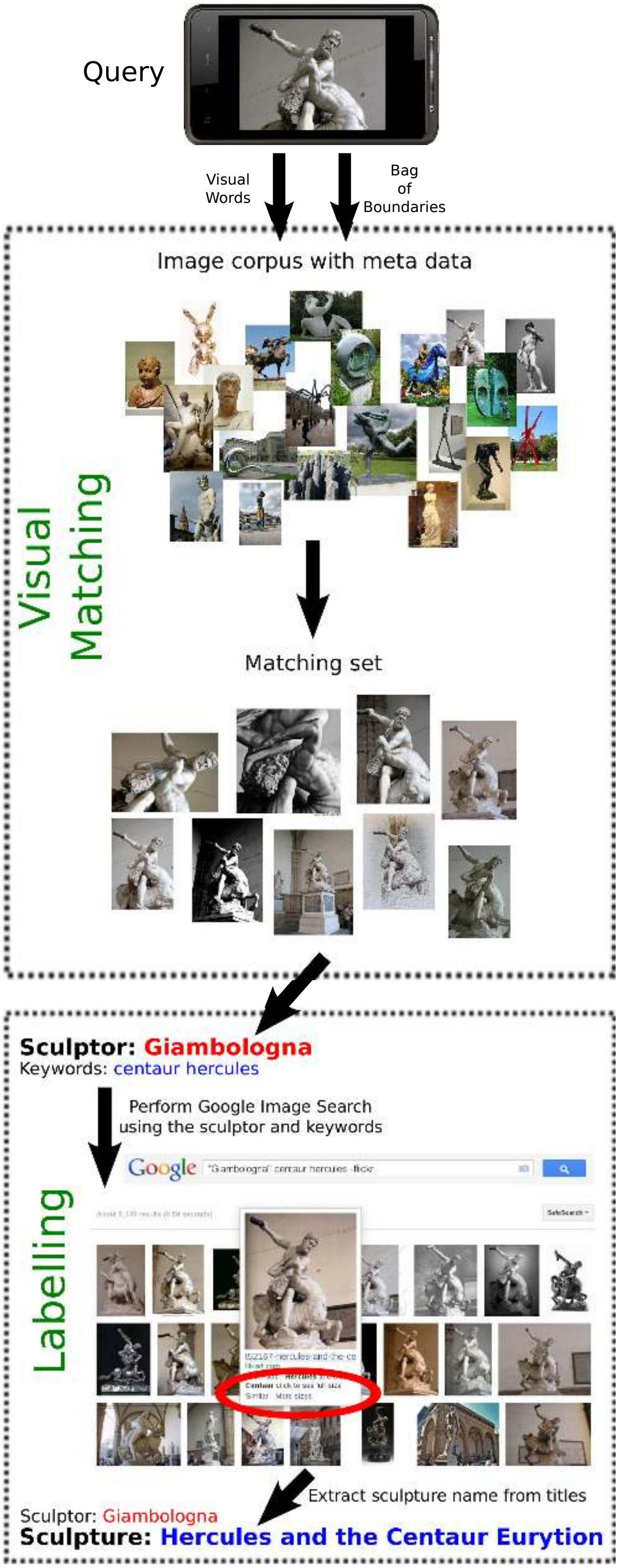
Query
Visual Words
Bag of Boundaries
Image corpus with meta data
Visual Matching
Matching set
Sculptor: Giambologna
Keywords: centaur hercules
Perform Google Image Search using the sculptor and keywords
Google
Labelling
Extract sculpture name from titles
Sculptor: Giambologna
Sculpture: Hercules and the Centaur Eurytion

Figure 1: Sculptor and sculpture identification: on-line system overview.

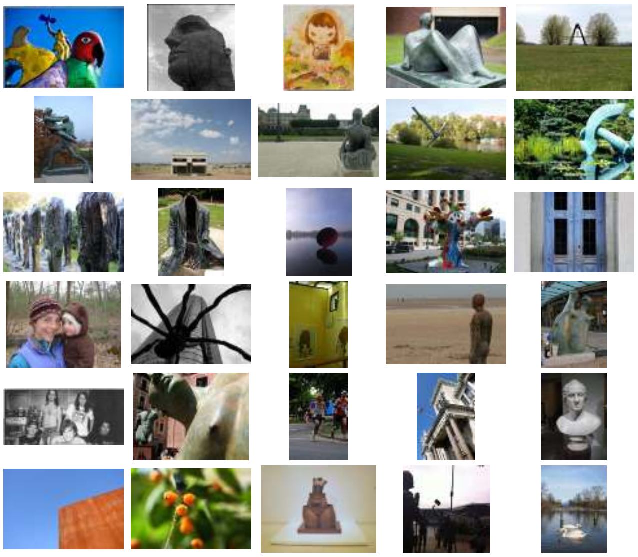

Figure 2: Random sample from the Sculptures 50k dataset.

mented. In each case, a visual query is used as the input and the system returns a ranked list of matched images from the dataset, where the ranking is based on the number of spatial correspondences between the query and target image. The outputs of these two systems are then combined as described in section 3.3.

3.1 Review of visual word large scale retrieval

The method proceeds in the following steps: First, each image in the corpus is processed to detect interest points/regions and these are represented using RootSIFT descriptors [5]. These descriptors are then vector quantized to map every feature to a "visual word". Each image is then represented by the histogram of visual words that it contains – a Bag of (visual) Words (BoW), and an index built to enable fast retrieval.

At query time the system is presented with a query in the form of an image region. This region is itself processed to extract feature descriptors that are mapped onto the visual word vocabulary, and these words are used to query the index. Retrieval then proceeds in two steps: an initial ranking/short list based on common visual words [17, 20] and tf-idf weighting; followed by a reranking based on a geometric relation to ensure that the query and target have compatible spatial configurations. In order for the retrieval to be immediate both steps must be extremely efficient at run time. The visual words enable text retrieval like scalability and speed, but the reranking is potentially more costly as it involves computing a geometric relation between the query image and each target image in the short list. The geometric relation of choice is an affine transformation [13, 18], which is computed using a robust estimation algorithm, and the images in the shortlist are then reranked by the number of matches between the query and target image that are consistent with the computed transformation.

We use a standard BoW retrieval system described in [18] with affine-Hessian interest points [15], RootSIFT descriptors [5], a vocabulary of 1M vision words obtained using approximate k-means, and spatial re-ranking of the top 200 tf-idf results using an affine transformation. The system returns a ranked list of images with each scored by the number of features (visual words) matched to the query.

3.2 Review of boundary descriptor large scale retrieval

The boundary descriptor retrieval method [4] follows some of the elements of the visual word retrieval system, in that an inverted index is used on quantized descriptors, but instead of representing the entire image only the boundaries of certain objects are represented. This quantized boundary descriptor – the bag of boundaries (BoB) representation – is computed and stored for each image in the corpus. Then for a query image, images from the corpus are retrieved in a similar manner to the visual word system by first ranking on the similarity of the BoB descriptor, and then reranking a short list by the spatial compatibility between the query and retrieved image using an affine transformation computed on local boundary descriptors.

The BoB representation is obtained in three stages: first, 'relevant' objects are segmented automatically to suppress background clutter in the image; second, their boundaries are described locally and at multiple scales; and, third, the boundary descriptors are vector quantized and the BoB representation is the histogram of their occurrences.

The segmentation stage involves an over-segmentation of the image into regions (super-pixels) using the method of [6], followed by classification of the super-pixel into foreground (sculpture) or background. The classifier is learnt from a training set where various sculptures have been annotated as foreground. The details are given in [4]. We use the same feature vectors and training set (which is provided on-line). Note, in [4] the segmentation is learnt and applied predominantly to sculptures of Henry Moore and Rodin. However, we have found that the trained classifier performs well over a substantial proportion of the images in the *Sculptures 50k* dataset, even though these contain a vast variety of different sculptures.

Descriptors are computed by sampling the object boundaries (internal and external) at regular intervals in the manner of [8] and computing a HoG [9] vector at each sample. 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 HoG descriptors are computed at multiple scales relative to the size of the foreground segmentation. The boundary descriptors are vector quantized using k-means. This is similar to the 'shapeme' descriptor of [16].

The spatial compatibility between the query and target image is determined by computing an affine transformation between the quantized boundary descriptors. The method returns a ranked list of images with each scored by the number of descriptors matched to the query.

3.3 Combining retrieval results

The two described methods are complementary, BoW is well suited for retrieval of textured objects while BoB is adapted for smooth textureless objects defined mainly by their shape. Often only one of the systems is appropriate for a particular sculpture/query, but in a significant number of cases both systems manage to retrieve correct results; each of the cases can be observed in figure 4.

The two systems are combined by scoring each image by the maximum of the individual scores obtained from the two systems. In the situation where one system is capable of handling the query and the other is not, the latter system assigns low scores to all images while the former sets high scores to the relevant ones; the max combination rule thus correctly retrieves the relevant images by trusting the high scores of the former system. Note that our combination method merges the results of the two systems softly, i.e. no hard decisions are made about which system should be solely trusted for a particular query. This is because for some queries both systems are capable of functioning correctly, and the union of their matching sets (automatically obtained by the max combination method) is larger than any of the individual matching sets. We have not found it necessary to calibrate the scores obtained from the two systems.

The output of this stage is a *matching image set* which contains the highest ranked images of the combined list with score above a threshold of nine. Each image has an associated matching score and also the meta-data originally obtained from Flickr. This choice of threshold retains only the most reliable matches to the query. If this procedure yields no results, then the top match (highest ranked) is returned with low confidence.

4. SCULPTOR AND SCULPTURE IDENTIFICATION

The goal of this work is to create a system which can automatically determine the sculptor and sculpture in a query photo. This is done by querying the Sculptures 50k database with the image, as described in section 3, and processing the textual meta information associated with the matching image set. Here we describe how to obtain the sculptor and sculpture names from this set of meta data.

4.1 Sculptor identification

It is simple to propose a candidate sculptor for each retrieved image: it is sufficient just to look up the textual query (i.e. sculptor name, see section 2) which was used to download that image from Flickr when the Sculptures 50k database was harvested. Given this set of putative sculptors we propose and compare two simple strategies to identify the actual sculptor of the query image: *Winner-Takes-All* and *Weighted Voting.*

Winner-Takes-All (WTA). The top match (highest ranked) is kept as the correct one and its sculptor identity is returned. Empirically, this scheme performs quite well, however it does have two shortfalls: it is prone to *label noise* and *retrieval failure.* In the *label noise* failure case the system cannot identify the sculptor correctly due to the mislabelled top match, which is a significant possibility when data is obtained in an unconstrained fashion, in our case from Flickr user annotations. *Retrieval failure* occurs if the top match is not actually of the same sculpture as the query. Both of these can be overcome to some extent by the following scheme.

Weighted Voting (WV). The scores of the top four images in the matching set are counted as weighted votes for the sculptor associated with that image; the sculptor with the largest sum of votes is chosen. This method can sometimes overcome both failure cases of the WTA scheme (label noise and retrieval failure) if the database contains more than one image of the same sculpture and they are correctly retrieved. As shown in section 5, this strategy outperforms Winner-Takes-All by 2%.

4.2 Sculpture identification

Unlike identifying the sculp*tor*, identifying the sculp*ture* requires finding distinctive words in the textual meta data associated with the matching image set. However, this data is variable, unstructured and quite noisy as it is supplied by Flickr users so it needs to be cleaned up and normalized. We first describe the filtering procedure that is performed off-line for data clean-up, and then the on-line sculpture naming applied to the matching image set.

1. Off-line: Meta data preprocessing. The data is cleaned up and normalized by the following steps. First, to reduce the problem of having different languages in the meta data, Microsoft's automatic translation API is used to detect the language and translate the text into English. This procedure overcomes sculptures being named in different languages, e.g. Rodin's "The Thinker" is also commonly referred to as "Le Penseur" in (the original) French.

Second, characters such as ,;:_-&/\()@ are treated as spaces in order to simplify word extraction and standardize the text. For example, Henry Moore's sculpture "Large Upright Internal External Form" contains the words "Internal External" which are also often found in variations such as "Internal-External" or "Internal/external"; all these cases are identical after the standardization step.

Only alphanumeric characters are kept and converted to lower case in order to simplify word correspondence. Of these, only words longer than 2 and shorter than 15 are kept so that typos, abbreviations and invalid words are filtered out. Some uninformative words are removed too, like "Wikipedia", "Wiki", "www", "com", "jpg", "img" etc. Also, only words which do not contain any digits are kept in order to filter out image file names often found in Flickr meta data, such as DSC12345, IMG12345, P12345, as well as the dates the photos were taken.

Lastly, the name of the sculptor is removed from the meta data in order to enable sculpture identification instead of just obtaining the sculptor name again.

2. On-line: sculpture naming. We start here with the meta-data associated with the matching image set for the query image. Only the meta-data from the images with the previously identified sculptor (section 4.1) is used in order to filter out potentially falsely retrieved images (i.e. those images that were in the original matching set, but do not contain the sculptor name selected by WTA or WV). There are two steps: (i) first, keywords, often containing the name or a part of the name, are identified, and second, the name is verified or corrected using Google by a form of query expansion.

The sculpture name, or particular words which can be used to uniquely identify the sculpture, are obtained by finding words which frequently occur in the titles and descriptions of the matching set, but are distinctive at the same time (for example typical stop-words such as "the" are common but

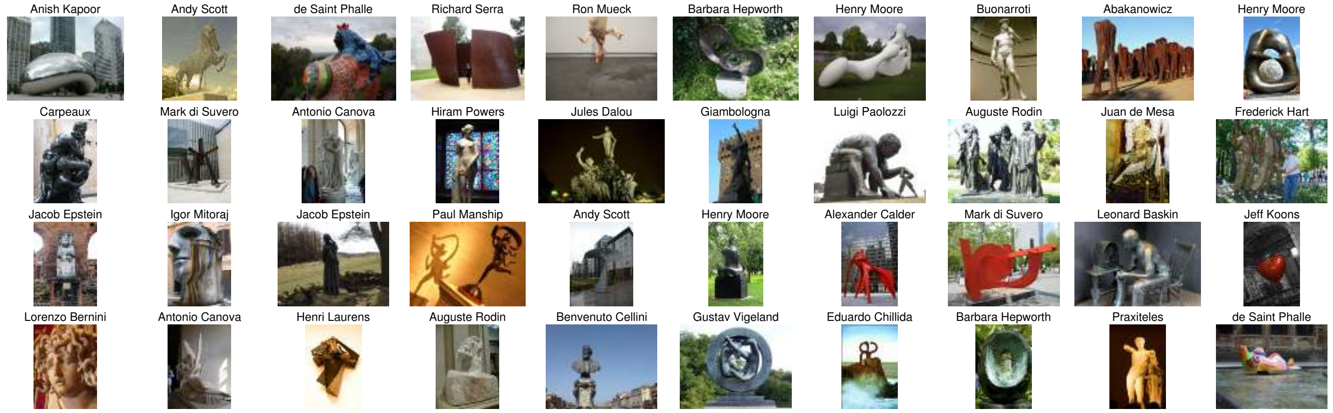

Figure 3: Random sample of evaluation query images (40 out of 200) used for evaluation, and sculptor names for each image (first names for some sculptors are cropped for display).

not at all distinctive). This is achieved by the standard *term frequency inverse document frequency (tf-idf)* [7] weighting, where each word is weighted by its frequency in the text and down-weighted by the logarithm of the number of documents in the entire database which contain it. We term the top scoring words *keywords*, and these identify the sculpture and are most commonly part of its name. However, further processing is required to obtain the full sculpture name from these keywords, for example it is required to put the words in the right order, add non-distinctive words as many sculpture names contain words like "the" and "and", and correct possible mistakes.

Google based query expansion. The top two scoring keywords from the previous step, along with the name of the sculptor are used to issue a textual query on Google image search; the titles associated with the top Google image search results are then processed, as described next, to obtain the full sculpture name. The procedure is illustrated in figure 1, where the sculptor is identified as "Giambologna" and the top two keywords as "centaur" and "hercules", resulting in a Google image search query *"Giambologna" centaur hercules -flickr* ("-flickr" is added to obtain results independent of the Sculptures 50k dataset which is entirely downloaded from Flickr). The textual data associated with the top Google image search results is mined to obtain the full sculpture name "Hercules and the Centaur Eurytion".

The full sculpture name is obtained by "growing" the top scoring keyword using the titles associated with the top 15 Google image search results obtained via the Google API (note, only the titles from the Google image search API are used; the images themselves are not processed). The name is "grown" as follows: firstly, it is initialized to be the top keyword. Secondly, the name is iteratively expanded by the word which directly precedes or succeeds it in the most number of titles. In our running example (figure 1) the initial name is "centaur", the word "the" directly precedes it 8 times, "a", "with" and "beating" once each, and it is succeeded by "Nessus" trice and "Eurytion" twice; "the" thus has most support and is prefixed to the name to form the new one: "the centaur". Words are prefixed or appended to the name one by one; growing stops once a proposed name does not exist in more than one title, in order not to grow it to an entire title and overfit. The procedure yields many benefits: the name length is automatically determined (otherwise one would need to set a threshold on the tf-idf scores which is potentially hard to do), words are put in the correct order, new non-distinctive words like "the" and "and" which have a low tf-idf score are automatically inserted to form a meaningful sequence of words, etc.

5. EVALUATION AND RESULTS

The on-line system including all the stages takes only 0.8s from specifying the query image to returning the sculptor and sculpture name on a standard multi-threaded 2.8 GHz workstation. The memory required to store the (non-compressed) files for the retrieval system is 4.3 Gb. Of the run time, on average, 0.56s is used for visual matching, 0.23s for calling the Google API, and 2ms for sculpture and sculptor naming.

To evaluate the identification performance we have randomly selected 200 images of various sculptures from the Sculptures 50k dataset. For each of these we have manually confirmed that at least one more image of the same sculpture exists in the database, as otherwise identification would be impossible. A sample of these is shown in figure 3, illustrating the wide variety of sculpture images and sculptors used for evaluation.

The system can be evaluated at three levels: visual matching, sculpt*or* naming and sculpt*ure* naming, and we report on each of these in turn.

5.1 Visual matching

Visual retrieval failures happen due to well known problems like extreme lighting conditions, inability to handle wiry objects, large segmentation failures (BoB method), interest point detector dropouts (BoW method), descriptor quantization etc. Segmentation can fail when the sculpture is not isolated from the background physically, for example draped with a sheet.

The visual matching is quantitatively evaluated here by reporting the proportion of queries for which the query sculpture appears in the top four retrieved results. Note that we define the "same sculpture" relationship as it is defined in [4], namely: the two sculptures are the same if they have identical shapes, but they do not need to be the *same instance*, as the same sculpture can be produced multiple times (and can be made in different sizes and from different materials).

The BoB and BoW methods successfully retrieve the correct

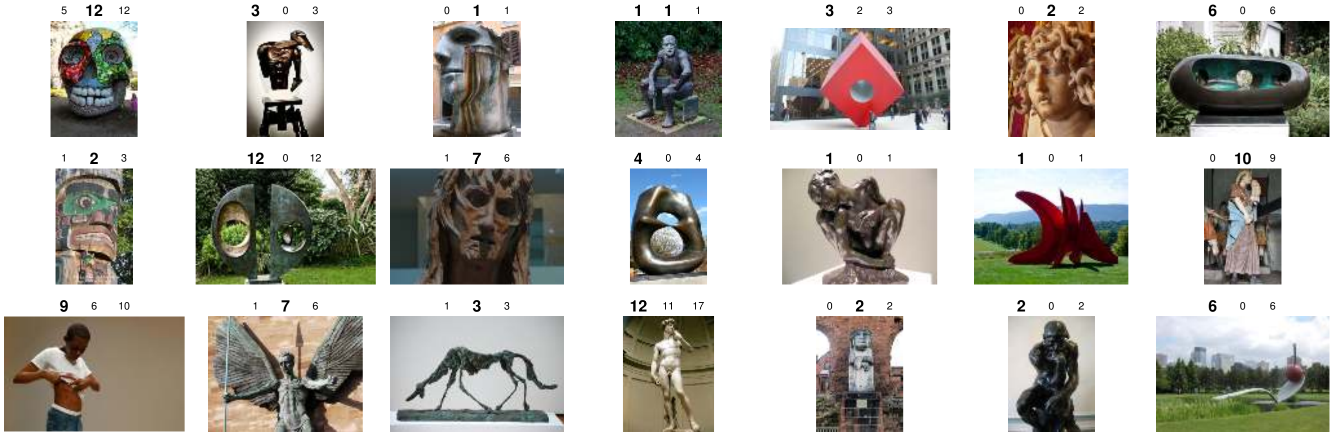

Figure 4: Comparison of BoB and BoW methods for 21 (out of the total of 200) evaluation query images. The numbers above each query image are the number of positives retrieved before the first negative for each of the methods; from left to right these are BoB, BoW and the combined method. A high number corresponds to better performance and indicates both that the first mistake was low ranked, and also that there are many examples of that sculpture in the *50k* dataset. Numbers shown in bold and larger font point out the better method to be used for the given image (BoB or BoW).

sculpture within the top four results 60.5% and 63.5% of the time, respectively, while the combined method achieves 83.5%. The large performance boost obtained by combining the two methods demonstrates that they capture complementary information. Figure 4 shows query images and performances of BoB, BoW and the combined method.

5.2 Sculptor identification

The performance measure for sculptor identification (section 4.1) is the proportion of times the retrieved sculptor matches the sculptor of the query image. Recall that images were downloaded from Flickr by issuing textual queries with sculptor names, so the image-sculptor association is automatically available but potentially noisy (i.e. may not be the true sculptor).

The *Winner-Takes-All (WTA)* scheme correctly identifies the sculptor 154 times, i.e. achieves the score of 0.770, while *Weighted Voting (WV)* achieves 0.785, i.e. WV succeeds 94% of the times that the visual matching is correct. Compared to WTA, WV manages to overcome three retrieval failures and two noisy labellings, while introducing one mistake and changing an accidental positive into a negative.

In the case of WV, the BoB and BoW methods achieve 0.550 and 0.635, respectively, while the combined method achieves 0.785. If we instead imagine that there is an oracle that decides which retrieval method to trust to obtain the matching set for each query image a performance of 0.835 is achievable.

5.3 Sculpture identification

It is harder to evaluate sculpture identification (section 4.2) as sculptures often do not have distinctive names (e.g. many Henry Moore's sculptures are known simply as "Reclining Figure"), and query image descriptions are too noisy to be used as ground truth (unlike the case of sculptor naming, there is not a simple image-sculpture association available from the query used to download the data). As a proxy for evaluating the obtained sculpture names we perform a textual Google image search and check if an image of the queried sculpture is present in the top 15 results. We have manually done this evaluation for each of the 200 queries and recorded the proportion of times a hit was obtained.

The Google image search query is formed by concatenating the sculptor name (surrounded by citation marks), followed by the top two keywords obtained in the procedure from section 4.2, appended by "-flickr" in order to make sure we do not simply retrieve back images from our dataset as the text would certainly match. For the example shown in figure 1, the system returns "Giambologna" as the sculptor and the top words are "centaur" and "hercules", then the Google image search query used to evaluate these results is *"Giambologna" centaur hercules -flickr*. Note that the query string is identical to the one used for query expansion. The obtained search results (also shown in figure 1) contain many examples of the queried sculpture thus confirming identification success.

The combined method achieves a sculpture identification score of 0.615. This means that it succeeds 78% of the times that the sculptor identification is correct. Unlike other Flickr annotations we have found the annotations of sculpture images to be fairly reliable. For this reason, it has not been necessary to go to further efforts in overcoming noisy annotations such as [14, 21]. Qualitative examples in figure 5 demonstrate the effectiveness of our method.

The meta data clean up and normalization step (section 4.2) is very important since switching off the described preprocessing (while still using automatic translation and removing the sculptor name) causes the performance to drop by 19%, to 0.500. Even when identification still succeeds, the proportion of correct images retrieved in the top 15 Google image search results substantially decreases, the obtained keywords are much noisier and full sculpture names are substantially worse.

Meaningful name extraction. The procedure used to obtain the full sculpture name from identified keywords (section 4.2) has been found to work very well. The keywords are successfully put in order, for example the top two key-

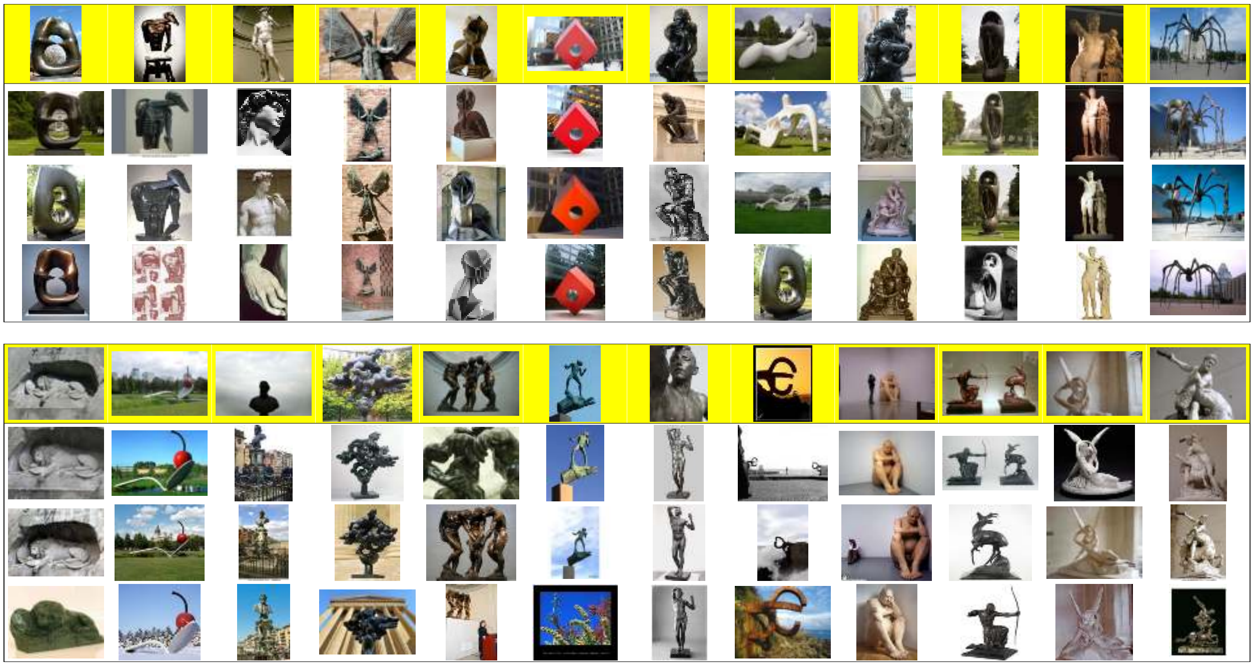

Figure 5: Examples of sculpture naming evaluation. Illustrations are laid out in two four-row blocks, each column in one block shows one example. For each example the top row shows the query image highlighted in yellow, while the remaining three rows show the top three Google image search results (section 5.2) using the identified keywords as textual queries (section 4.2). A wide variety of sculptures are correctly identified.

words “thinker the” are converted into “The Thinker”, as well as grown into a meaningful sculpture name, for example the top two keywords “sons ugolino”, “vulture strangling”, “call arms”, “lion lucerne’ and’ “rape sabine” are automatically and correctly converted into “Ugolino and His Sons”, “Prometheus Strangling the Vulture II”, “The Call to Arms”, “The Lion of Lucerne” and “The Rape of the Sabine Women”, respectively.

The fact that only the top keyword is used for name growing also means that mistakes from thresholding the keywords can be corrected. For example, the top two keywords for Michelangelo’s “David” are “david” and “max”, where the latter keyword is meaningless since the sculpture has a one-word title. The name growing procedure starts from “david” and stops at the very beginning correctly yielding “David”, as no expansion was found with sufficient support. Finally, it is worth noting that the Google image search using an automatically generated textual query can also flag a failure when the search yields very few or no results.

Actual outputs of the full system on a subset of the evaluation queries are shown in figures 1 and 6. The complete set of results over all 200 evaluation queries are provided online [3].

Failure analysis. Here we concentrate on problems related to sculpture naming given successful visual retrieval and sculptor naming.

(i) Bad annotation: The top retrieved results contain false or irrelevant annotation, or no annotation at all, rendering identification impossible.

(ii) Place domination: The textual description is dominated by the sculpture location thus still potentially correctly specifying the sculpture but not providing a useful

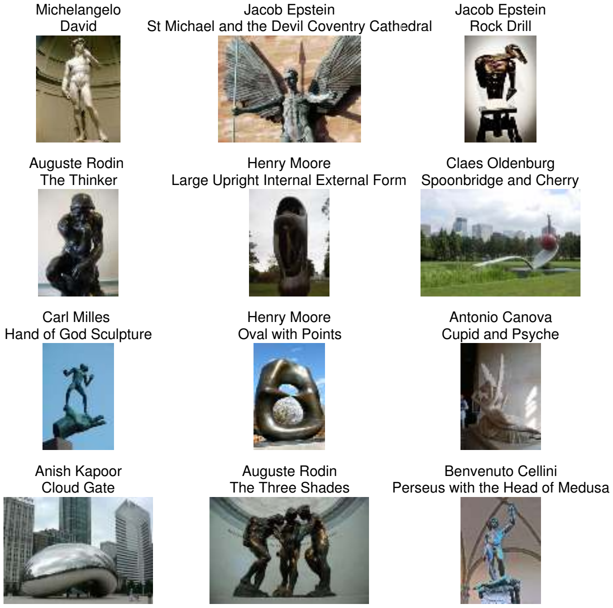

Figure 6: Sculpture naming examples. Automatically obtained sculptor and sculpture name are shown above each evaluation query image.

name for it; examples include names of museums or sculpture parks. This issue does not necessarily cause a failure since the information is often enough to uniquely identify a sculpture, for example: the top two words found by our method for Jacob Epstein's "St Michael and the Devil" in Coventry are "coventry" and "michael", all top 15 results from Google image search show the same sculpture.

(iii) Rare words dominate: Sometimes rare words, such as spelling errors, slang, unusual names etc, can dominate the results as they are deemed to be highly informative. On the other hand, the sculpture "We will" by Richard Hunt fails to be identified as both words are very common.

(iv) Name lost in translation: In this case the name of the sculpture is most widely known in its original form, thus performing Google image search for its English translation fails to retrieve relevant results, even though the sculpture identity has been effectively correctly identified. In our 200 evaluation queries we haven't noticed catastrophic failures due to this problem, however it is possible it would occasionally prevent identification. One example in which the interference is significant, but not sufficient for identification to fail, is in the case of Joan Miró's "Woman and Bird" (original Catalan: "Dona i Ocell"); where the top two words are correctly identified as "woman" and "bird" yielding only 5 out of 15 correct top Google image search results, while searching for the original words "dona" and "ocell" gives 14.

(v) Translation mistakes: Automatic translation fails to detect the correct language and/or to translate the text correctly into English. These are not necessarily catastrophic and in many cases the correct answer was obtained despite these failures (for example Rodin's "Kiss" is identified as "Kiss" and, in (the original) French, "Baiser").

(vi) Finally, our choice of evaluation measure can sometimes be a source of false negatives. For example, Gatzon Borglum's monumental sculpture "Mount Rushmore" is correctly identified by our method, but searching on Google images for *"Gatzon Borglum" mount rushmore* mostly yields images of the sculptor with image descriptions such as "Gatzon Borglum, the sculptor of Mount Rushmore".

In the light of this failure analysis and the noisiness of Flickr annotations, the achieved sculpture identification score of 0.615 demonstrates that the system really performs quite well.

6. CONCLUSIONS AND FUTURE WORK

We have demonstrated that sculptors and, with somewhat less success, sculptures can be named given a query image of a particular sculpture.

The next stage is to scale up the dataset further as having more examples of each sculpture in the corpus will overcome many of the failure cases of the sculpture naming. One avenue we are investigating is adding an authority score depending on the origin of the photo and text, e.g. the meta-data could have more authority if the photo is from Wikipedia rather than Google Image search or Flickr; or more authority if sculptures match when contributed by several different Flickr sources.

Acknowledgements. We are grateful for financial support from ERC grant VisRec no. 228180.

References

[1] http://en.wikipedia.org/wiki/list_of_sculptors.

[2] http://www.flickr.com/.

[3] http://www.robots.ox.ac.uk/∼vgg/research/sculptures/.

[4] R. Arandjelović and A. Zisserman. Smooth object retrieval using a bag of boundaries. In *Proc. ICCV*, 2011.

[5] R. Arandjelović and A. Zisserman. Three things everyone should know to improve object retrieval. In *Proc. CVPR*, 2012.

[6] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. From contours to regions: An empirical evaluation. In *Proc. CVPR*, 2009.

[7] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval.* ACM Press, ISBN: 020139829, 1999.

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

[9] N. Dalal and B. Triggs. Histogram of Oriented Gradients for Human Detection. In *Proc. CVPR*, 2005.

[10] S. Gammeter, L. Bossard, T. Quack, and L. Van Gool. I know what you did last summer: object-level auto-annotation of holiday snaps. In *Proc. ICCV*, 2009.

[11] Hartmut Neven, Google. Machine learning in Google Goggles. In *keynote talk, ICML, http://techtalks.tv/talks/54457/*, 2011.

[12] I. Ivanov, P. Vajda, L. Goldmann, J.-S. Lee, and T. Ebrahimi. Object-based tag propagation for semi-automatic annotation of images. In *ACM Multimedia information retrieval*, 2010.

[13] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Proc. ECCV*, 2008.

[14] Y. Jin, L. Khan, L. Wang, and M. Awad. Image annotations by combining multiple evidence and wordnet. In *ACM Multimedia*, 2005.

[15] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *IJCV*, 1(60):63–86, 2004.

[16] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. *IEEE PAMI*, 2005.

[17] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. CVPR*, 2006.

[18] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proc. CVPR*, 2007.

[19] T. Quack, B. Leibe, and L. Van Gool. World-scale mining of objects and events from community photo collections. In *Proc. CIVR*, 2008.

[20] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, 2003.

[21] C. Wang, F. Jing, L. Zhang, and H. Zhang. Image annotation refinement using random walk with restarts. In *ACM Multimedia*, 2006.