A Performance Evaluation of Gradient Field HOG Descriptor for Sketch Based Image Retrieval

Rui Hu[a,*], John Collomosse[a]

[a]*Centre for Vision, Speech and Signal Processing, University of Surrey, UK.*

Abstract

We present an image retrieval system for the interactive search of photo collections using free-hand sketches depicting shape. We describe Gradient Field HOG (GF-HOG); an adapted form of the HOG descriptor suitable for sketch based image retrieval (SBIR). We incorporate GF-HOG into a Bag of Visual Words (BoVW) retrieval framework, and demonstrate how this combination may be harnessed both for robust SBIR, and for localizing sketched objects within an image. We evaluate over a large Flickr sourced dataset comprising 33 shape categories, using queries from 10 non-expert sketchers. We compare GF-HOG against state-of-the-art descriptors with common distance measures and language models for image retrieval, and explore how affine deformation of the sketch impacts search performance. GF-HOG is shown to consistently outperform retrieval versus SIFT, multi-resolution HOG, Self Similarity, Shape Context and Structure Tensor. Further, we incorporate semantic keywords in to our GF-HOG system to enable the use of annotated sketches for image search. A novel graph-based measure of semantic similarity is proposed and two applications explored: semantic sketch based image retrieval and a semantic photo montage.

Keywords: Sketch Based Image Retrieval, Bag-of-Visual-Words, Image Descriptors, Matching.

1. Introduction

Digital image repositories are commonly indexed using manually annotated keyword tags that indicate the presence of salient objects or concepts. Text-based queries can concisely encode such *semantic* data, but can become cumbersome when representing *visual appearance* such as complex object shape — language concisely defines only a few stereotypical shapes. By contrast, visual queries such as free-hand sketches are an intuitive way to depict an object's appearance. Yet there has been comparatively little work exploring scalable solutions to sketch based image retrieval (SBIR) of photo collections which can also endure a certain degree of affine invariance.

A key challenge in SBIR is overcoming the ambiguity inherent in sketch. The casual, throw-away act of sketch query, coupled in some cases with limited user depictive skill, often results in sketches that depart from their target object — exhibiting variation in rotation, scale and translation within the scene and often with the embellishment or omission of strokes. Previous solutions have addressed this ambiguity by posing SBIR as model fitting approach [1]; aligning the sketch to underlying image data. This can be effective but involves a computationally expensive per-record optimization that also scales only linearly i.e. $O(n)$ with dataset size. Other solutions adopt a more traditional image retrieval pipeline, griding the image and concatenating descriptors from each cell to form a descriptor [2]. Such descriptors can be scaled sub-linearly with appropriate search structures (e.g. hierarchical search, or *kd*-trees), yet the advantages in speed conferred by the simplicity of image division are diluted by the lack of affine invariance inherent in that approach.

Our work adopts a Bag of Visual Words (BoVW) approach to SBIR, incorporating an adapted form of the Histogram of Oriented Gradients (HOG) descriptor; *Gradient Field HOG (GF-HOG)*. BoVW is considered to be the state of the art in Query by Visual Example (QVE) image retrieval approaches. Over the past few years, BoVW systems using photographic, rather than sketched, queries regularly exhibit leading performance and scalability in internationally recognized benchmark tasks (PASCAL, ImageCLEF [3]).

At its essence the BoVW framework for QVE focuses upon the identification of "visual words" within an image, which are derived from texture local to detected

*Corresponding author.

Email addresses: r.hu@surrey.ac.uk (Rui Hu), j.collomosse@surrey.ac.uk (John Collomosse)

interest points. The standard BoVW methodology is to construct a frequency histogram that represents a count of the visual words extracted from the image, and to use that frequency histogram as a global image descriptor in subsequent matching. Widespread success suggests that this non-spatial distribution of texture is sufficient for QVE using real imagery [4, 5]. Tailoring BoVW to SBIR is challenging in two respects: sketches contain little or no texture, and sketches are defined by the relative spatial positions of their constituent strokes. However, these are exactly the information abstracted away by BoVW.

In this paper we describe in detail the GF-HOG descriptor [6], and contribute a comprehensive evaluation of the performance of GF-HOG within a BoVW framework for SBIR. We perform a detailed evaluation of GF-HOG/BoVW performance, exploring various vocabulary sizes for BoVW and comparing against standard gradient binning descriptors for photo-based QVE (SIFT, HOG) and against approaches that have in the past been experimented for sketch based matching (Self-Similarity, Shape Context, Structure Tensor) within the BoVW framework. We also experiment with eight commonly used distance measures from norms to metrics frequently used in text ("Bag of Words") retrieval. The evaluation is performed over a dataset of around 15k Creative Commons licensed photographs from Flickr, containing 33 object categories and using 10 non-expert sketchers working "blind" i.e. drawing in a realistic retrieval scenario without tracing the target image, or otherwise having the target image in sight. A timing analysis of the descriptors is also presented. We also demonstrate a simple photo montage application that enables users to composit sketch-retrieved objects from our dataset. This application also demonstrates how GF-HOG may be combined with RANSAC to localize the sketched object in retrieved images.

A further novel contribution of this paper is the fusion of semantics and our GF-HOG shape description to enable scalable search using annotated sketches. Although sketch can be a complementary query mechanism to text for image retrieval systems, sketches alone are an inexact and ambiguous query specification. For example, a query depicting a ball (circle) might also return pictures of the moon and other circular objects. We tackle this problem by combining shape retrieval with a measure of semantic similarity based on tag co-occurrence. For example, this enables us to search for a circle, within the context of sports, or of space. The user annotated tags for Flickr images represent semantic information of the images. In this paper, a novel graph-based measure is proposed to compute the semantic similarity of keyword tags, which is fused with the content similarity. We demonstrate both semantic sketch retrieval (including a quantitative analysis of performance) and a photo montage system that enables sketch based composition of graphics, with each iteration sketch based search taking into account the semantic context of previously retrieved images within the montage.

2. Related Work

Sketch query based retrieval system has been used for searching trademarks [8, 9], technique drawings [10, 11], documents [12], hand drawn image set or clip arts [13, 14, 15]. These works often consider sketches as a combination of strokes or geometry units and the relation between them are used to represent the shape for matching [16, 17, 18]. However, these techniques can not be easily generalized to natural images.

The early nineties delivered several SBIR algorithms capable of matching photographs with queries comprising blobs of color, or predefined texture. These systems were based primarily on global color histograms [7], spatial patterns [19] or region adjacency [20, 21]. Robust color blob-based matching was later proposed using spectral descriptors, such as wavelets [22, 23] which have more recently proven useful for more general image retrieval using colour texture [24, 25].

Prior work, more closely aligned with our proposed approach, has explored the sketching of line-art shape depictions for SBIR [26] using edge maps to approximate a sketch from a photograph prior to matching. Matusiak *et al.* [27] applied curvature scale space (CSS) [28] as a robust contour representation, although this required a pre-process to first extract the contour. By contrast, the majority of early sketched shape retrieval work focussed on the fitting of sketched contours directly to the image. Bimbo and Pala [29] present a technique which based on elastic matching of sketched templates over edge maps in the image to evaluate similarity. The degree of matching achieved and the elastic deformation energy spent by the sketch to achieve such a match are used to derive a measure of similarity between the sketch and the images in the database and to rank images to be displayed. However the expense of the optimization step to fit the model limits the scalability of the approach, making it inappropriate for use in interactive SBIR systems such as ours (Fig. 1). Ip *et al.* [30] develop an affine invariant contour descriptor for SBIR, based on the convex hull of the contour and the curvature of a set of dominant points along the contour. Edge orientation [31, 32, 33, 8] and angular partitioning [34] have also been used to describe contours.

Later approaches sought to combine global descriptors for color (e.g. RGB histogram) with shape (e.g. edge orientation histogram) as a depiction invariant similarity measure for trademark image retrieval [31].

Chans *et al.* [32] tokenize edge segments into a string representation, encoding length, curvature, and relative spatial relationships. A similar approach was used for sky-line retrieval in [35]. Rajendran and Chang [33] propose to extract edge signatures for both images and sketch query, and compare their curvature and direction histograms for shape similarity. This method utilizes multiple scales to account for various levels of detail in sketch query. Shih and Chen [8] extract features which include invariant moments, the histogram of edge directions, and two kinds of transform coefficients that are robust to geometric deformation, to describe representative objects in the task of trademark segmentation and retrieval. Chalechale *et al.* [34] employ angular-spatial distribution of pixels in the abstract images to extract features using the Fourier transform. The extracted features are rotation and scale invariant and robust against translation. Eitz *et al.* [2] propose a real-time SBIR system that locates photos using a sketched depiction of object shape. Eitz *et al.* divide the image into a regular grid and compute descriptors (either EHD [8] or the Structure Tensor [2]) from each cell — concatenating these to form a global image feature. However, as we later show, this offers limited invariance to changes in position, scale or orientation compared to our approach and a number of other descriptors (e.g. Self Similarity [36]) within a BoVW system. A scalable approach to SBIR was presented very recently in MindFinder [37] through spatial gridding of edge fragments (edgels) detected within the image. The method is efficient, though as with [2] exhibits limited robustness to translation assuming sketches to approximately occupy the same location as the objects they depict.

Local descriptor matching using BoVW offers solutions to affine invariance and first applied to sketch retrieval by the authors in [6], where the GF-HOG descriptor was also proposed as an adaptation of HOG amenable for SBIR in the BoVW framework. GF-HOG is described in more detail within Section 3.2, and is a depiction invariant descriptor computed local to set pixels in the Canny edge map of an photo (or pixels of sketch strokes). Brief comparisons are made in [6] of GF-HOG to SIFT, HOG and Self Similarity over various vocabulary sizes — a study that we elaborate upon significantly in Section 6. Very recent work by Eitz *et al.* [38] also explores standard HOG within a BoVW framework; computing HOG at random points in the image, and in another experiment over pixels of a Canny edge map — the latter is dubbed S-HOG in [38] and is aligned with the definition of EDGE-HOG (multi-resolution HOG over Canny edges) used as a baseline for comparison in [6]. S-/EDGE-HOG is reported by both [38] and [6] to yield better performance than Shape Context-like features. Our evaluation compares S-HOG/EDGE-HOG to several other descriptors including standard HOG, SIFT, Self Similarity, as well as Shape Context. Furthermore, we explore a variety of distance metrics, the selection of which is known to significantly impact the outcome of a BoVW retrieval system [4, 3].

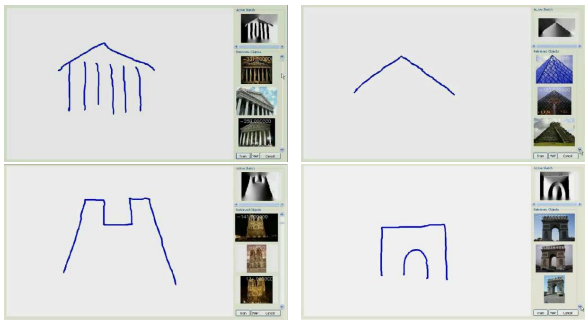

Figure 1: Our interactive SBIR system enables searches image collections for matching shapes using free-hand sketched queries by non-expert users.

Similar to our sketch based photo montage application, [39] Chen et al *et al.* propose Sketch2Photo — an interactive system in which keyword-annotated sketches are used to retrieve and composit photograph fragments. In that system, keywords trigger a Google Image search which returns possible images — the sketch is used only to crop the image using coarse shape matching via Shape Contexts. The sketched shape itself is not used for retrieval, nor is an integrated approach taken to fuse shape and text into a single similarity score as with our approach. Rather, the hybrid text-shape component of our approach is more closely aligned with the hybrid systems of Wang *et al.* the operation of which is outlined in [40, 41]. In [40] images are divided into local patches and matched with the sketch query. Similarly to [2] the matching has limited robustness of affine invariance. Tag similarity and content similarity is fused linearly, where tag is set a higher priority. While in [41], major curves of images are first detected, based on which a curve-based algorithm is conducted to achieve precise matching between sketch and image database. However, the technique detail is not given in the paper. The text and content models are worked as reranking conditions, where tag is the dominant search condition. In our system we leverage keyword co-occurrence, a text matrix widely used in information retrieval, and recommendation systems [42, 43] as well as in building word ontologies [44] and improving the image annotation performance [45]. Various methods of calculating keyword co-occurrence includes the Jaccard measure [46], the corrected Jaccard measure [46], measures from the information-theory [46] and the confidence

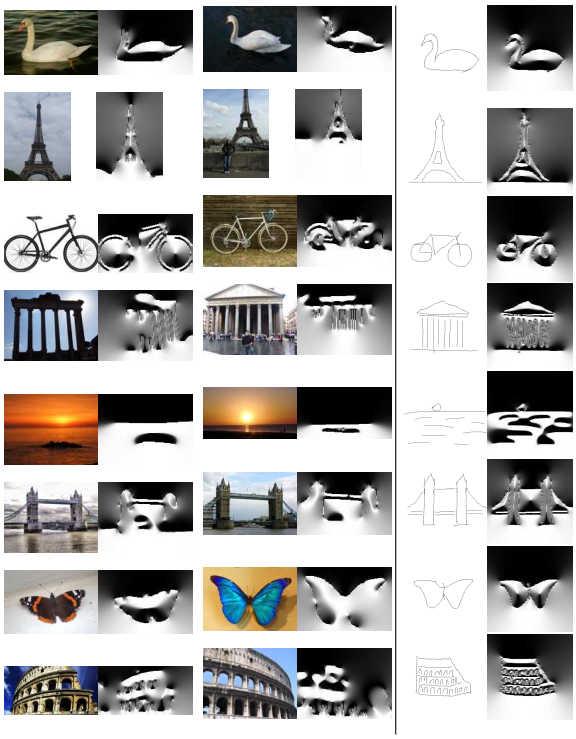

Figure 2: Visualizations of gradient field Θ derived from query sketches and corresponding photographs, via processing steps of subsec. 3.2.1.

score measure in association rule mining [47, 48, 42]. In this paper, we propose a graph based tag similarity measure which extents the asymmetric score measure of [42] so that the similarity between two tags considers not only the explicit co-occurrence but also the implicit co-occurrence of keywords. Note that our previous work [49] incorporating semantics for sketch based video retrieval. Annotated sketches are used as queries for video retrieval. Videos are first segmented into spatio-temporal volumes and a histogram distribution of the pixel-wise labelling for each volume is computed and matched with the annotation of the query sketch. The similarity of motion, color and semantics for each volume is merged by multiple the similarity scores. Since there is no tag available for the video dataset, semantic information are achieved by automatic pixel-wise labelling.

3. Gradient Field HOG Descriptor

In this section we briefly outline the Histogram of Oriented Gradients (HOG) descriptor, and how this is adapted to create our Gradient Field HOG (GF-HOG) descriptor.

3.1. Histogram of Oriented Gradients (HOG)

The HOG descriptor [50] is commonly applied in object recognition, and human (e.g. pedestrian) detection tasks. HOG is a window based descriptor computed local to a detected interest point (or in some cases densely sampled over all image points). The window is centred upon the point of interest and divided into a regular square grid ($n \times n$). Within each cell of the grid a frequency histogram is computed representing the distribution of edge orientations within the cell. The edge orientations are computed as $arctan(\frac{\delta I}{\delta y} / \frac{\delta I}{\delta x})$ and quantized into q bins. The histogram counts are concatenated to form a q-D vector for each cell, which are again concatenated to form an qn^2-D vector for the window. In many implementations, several windows are sampled in a non-overlapping $w \times w$ grid local to the key-point and again concatenated to output the final descriptor.

In our experiments of Section 6, we compute the HOG descriptor at multiple scales of $n = 5, 10, 15$ and fix $w = 3, q = 9$. The scale and window size are chosen to have a reasonable coverage of the objects in the images, since all the images are sampled to size between 100 to 200. These parameters are comparable to the recommended parameters (which we also use in this paper) for the descriptors we compare with in section 6. We do not compare this set of parameters to the others and pick the best set. The descriptors are pooled from multiple scales and used to construct a single codebook for BoVW. The interest points chosen to compute are local to detected Canny edges (in the case of photographs in the dataset) and local to pixels comprising sketched strokes (in the case of sketched queries). We note the practice of computing HOG over Canny edges is referred to as EDGE-HOG in [6] and S-HOG in [38].

3.2. Gradient field HOG

Sketches are characterized by the relative positions of their strokes, with little texture present (Fig. 6). To extract a window-based descriptor local to a sketched stroke, an appropriate window size should be chosen — large enough to encode encode local spatial structure in the neighborhood, but not so large as to capture too much content and lose discriminability. Automated size selection is non-trivial and could quickly become heuristic and tailored to particular content. We therefore adopt an more dynamic approach whereby the orientations of strokes (or Canny edges, in the case of photographs) are interpolated from nearby strokes under a Laplacian smoothness constraint, thereby labelling pixels that are not part of a stroke/edge and so have no meaningful orientation. In all cases a pixel's interpolated orientation is most strongly influenced by the closest strokes — so delivering a form of dynamic neighborhood selection.

3.2.1. Constructing the Gradient field

In order to encode the relative location and spatial orientation of sketches or Canny edges of images, we represent image structure using a dense gradient field interpolated from the sparse set of edge pixels.

We begin with a mask of edge pixels $M(x,y) = \{0,1\}$, derived either from the mask of sketched strokes or from the Canny edge map of a photograph. A sparse orientation field is computed from the gradient of these edge pixels:

$$\theta[x,y] \mapsto \arctan\left(\frac{\delta M}{\delta y} / \frac{\delta M}{\delta x}\right), \quad \forall_{x,y} M(x,y) = 1. \quad (1)$$

We seek a dense field Θ_Ω over image coordinates $\Omega \in \mathcal{R}^2$ constrained such that $\Theta(p) = \theta(p), \forall_{p\in\Omega} M(p) = 1$. The dense field should be smooth, and so we introduce a Laplacian constraint by seeking the Θ that minimizes:

$$\underset{\Theta}{\operatorname{argmin}} \int\int_\Omega (\nabla\Theta - \mathbf{v})^2 \quad s.t. \ \Theta|_{\delta\Omega} = \theta|_{\delta\Omega}. \quad (2)$$

which corresponds to the solution of the following Possion equation with Dirichlet boundary conditions

$$\triangle\Theta = \text{div}\mathbf{v} \text{ over } \mathbf{\Omega} \text{ s.t. } \Theta|_{\delta\mathbf{\Omega}} = \theta|_{\delta\mathbf{\Omega}} \quad (3)$$

where div is the divergence operator and $\mathbf{v}$ is the guidance field derived from the orientation field of the original image. A discrete solution was presented in [51], by forming a set of linear equations for non-edge pixels, that are fed into a sparse linear solver to obtain the complete field. Common applications such as image completion ("Poisson in-filling") approximate $\triangle\Theta = 0$ using a 3×3 Laplacian window: $4\Theta(x,y) = \Theta(x-1,y)+\Theta(x+1,y) + \Theta(x,y-1) + \Theta(x,y+1)$. However we obtained better results in our retrieval application when approximating $\triangle\Theta$ using a 5×5 window discretely sampling the Laplacian of Gaussian operator (leading to a smoother field):

$$\triangle\Theta(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}. \quad (4)$$

Images and sketches are padded with an empty border of 15% pixel area. For typical images of $\sim 200 \times 100$ the linear system is solvable in $\sim 100-150$ms using the TAUCS library on an Intel Pentium 4 [52]. Full timing analyses are given in subsec. 6.4. Fig. 2 illustrates the visual similarity between the gradient fields computed from corresponding sketches and photographs.

3.2.2. Multi-scale Histogram of Gradient

We compute a set of HOG-like descriptors over the gradient field by quantizing the field and establishing a $n \times n$ grid (via the process of subsec. 3.1) local to each point $M(x,y) = 1$ (i.e. pixels comprising sketched strokes, or in the case of database images, Canny edges). We detect features with $n = \{5, 10, 15\}$ and fix $w = 3$, $q = 9$, yielding several hundred Gradient Field HOG (GF-HOG) descriptors for a typical image. The use of multiple scales is reminiscent of Pyramid HOG (P-HOG) [53] but note that here we do not concatenate descriptors at each scale to form a feature — rather we add GF-HOG at each scale (n) to the image descriptor set.

4. Sketch Based Image Retrieval and Localization

The 'bag of words' model, first proposed for text document retrieval, is now well established as a framework for scalable image retrieval[4, 5, 54, 55]; so called 'bag of visual words (BoVW)'. In a typical BoVW framework interest points are first detected [56] and represented by descriptors [57]. Unsupervised grouping is performed over all descriptors, to generate a set of k clusters. Each cluster is a 'visual word', and the codebook of visual words comprises the vocabulary of the system.

In our system, the GF-HOG features are extracted local to pixels of the Canny edge map. Features from all images are clustered off-line to form a single BoVW codebook via k-means (section. 6.3 presents results for varying k). A frequency histogram H^I is constructed for each image, representing the distribution of GF-HOG derived visual words present in that image. The histogram is then normalized. At query time, a frequency histogram H^s is constructed from the query sketch by quantizing GF-HOG extracted from the sketch using the same codebook, and constructing a normalized frequency histogram of visual words present. Images are ranked according to histogram distance $d(H^I, H^s)$ as defined using one of the measures below.

4.1. Histogram distance measures

The distance measure between two frequency histograms is often critical to the success of BoVW search — and different similarity measures as cited as optimal for different domains (e.g. activity recognition [58], building search [5]). We briefly describe several most widely used distance measures in table 1 to compute the similarity between histogram of the query sketch to those of the images in the dataset. These are the most commonly used distance measures. H^I and H^s denote the sketch and candidate image histograms respectively. $H(i)$ indicates the normalized count of the i^{th} bin, where $i = \{1, ..., k\}$. The Mean Average Precision (MAP) result curves of each distance measure are compared in Sec. 6 to justify a choice.

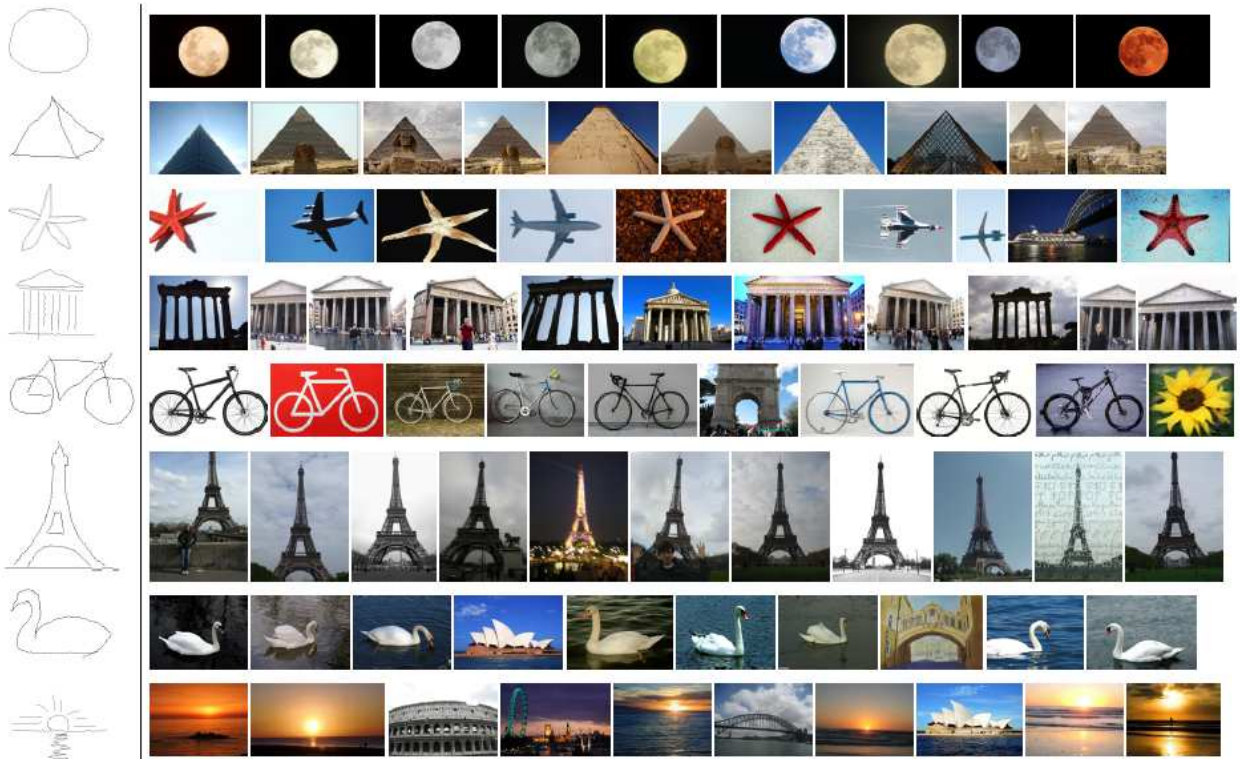

Figure 3: Example query sketch, and their top ranking results (ranking from left to right) over the Flickr15K dataset. Results produced using our GF-HOG descriptor in a Bag of Visual Words (BoVW) framework with vocabulary $k = 3500$ and histogram intersection distance.

Table 1: Distance measures

Cityblock	$\sum_{i=1}^{k} \lvert H^I(i) - H^S(i) \rvert$
Cosine	$1 - \frac{(H^I)^T H^S}{\lvert H^I \rvert \lvert H^S \rvert)}$
χ^2	$\sum_{i=1}^{k} (\frac{(H^I(i) - H^S(i))^2}{((H^I(i) + H^S(i)))}$
Histogram Intersection	$\sum_{i=1}^{k} \sum_{j=1}^{k} (\omega_{ij} \min(H^S(i), H^I(j))_i)$, $\omega_{ij} = 1 - \lvert \mathcal{H}(i) - \mathcal{H}(j) \rvert$, where $\mathcal{H}(i)$ is the visual word of the i^{th} bin.

4.2. Language models for retrieval

Following recent trends [5, 54, 59] we exploit the text retrieval lineage of BoVW by evaluating language models commonly used for text retrieval. These are implemented via the Lemur text retrieval framework [60], by synthesizing a text document containing a set of arbitrary but tokens for each visual word present in a database image or sketch. Lemur was then invoked to compute the distance between a query document and a candidate image document. Given a query sketch represented by histogram H^S, and the database of N images represented by histograms $H_n^I, n = \{1, ..., N\}$. The following models are used in our system, which are the most widely used language models in multimedia retrieval.

Term weighting (tf-idf). Term weighting computes a distance affording greater weight to rarer visual words (terms), within the context of the image collection. Term frequency (*tf*) is the ratio of a term's (i.e. visual word's) frequency over the others within an image. Document frequency *df* measures the frequency a term appears across all document. The inverse document frequency *idf* evaluates the rarity i.e. importance of the visual word $idf(i) = log(\frac{1}{df(i)})$. The *tf-idf* measure is then a product of a term's frequency and importance.

Okapi-BM25. BM25 is a family of scoring functions that take into account not only *tf-idf* but also the length (i.e. number of visual words) present.

InQuery. The InQuery (CORI) algorithm is based a cascade of inference networks trained over the frequencies of visual words present in the image collection using multiple regression passes. The similarity of H^I and H^S are evaluated by presenting the histograms as input to the trained network; [61] contains further details.

Kullback-Leibler (K-L) divergence. Similar in spirit to histogram intersection, K-L divergence is commonly

used to measure the difference between pairs of probability distributions in information theory, and is here used to measure the similarity between H^I and H^S: $d_{\mathrm{KL}}(H^S, H^I) = \sum_{i=1}^{n} H^S(i) log \frac{H^S(i)}{H^I(i)}$.

4.3. Identifying sketched object position

In order to present retrieval results, and for our prototype photo montage application (in section 5.3.2), it is desirable to identify the location of the sketched query within the retrieved images.

Given a sketch and a retrieved image, we create a set of putative correspondences between the GF-HOG descriptors cut from both the sketch and the image. These are matched in a nearest-neighbor manner within the descriptor space via Lowe's SIFT matching scheme[62] i.e. if descriptor A in a sketch is assigned to B in the image, then B must also be nearest to A for a valid match [62]. To preserve affine invariance, no spatial information e.g. proximity is used when matching. The repeatability of GF-HOG between sketches and photos (Fig. 2) promotes good matches despite absence of texture, yet the local similarity of edges may limit matching or create erroneous correspondences.

Typically a sketch is a mental abstraction of a shape, and so any attempt at localization is likely to be an approximation. We apply Random Sampling and Consensus (RANSAC) to fit the sketched shape to the image via a constrained affine transformation with four degrees of freedom (uniform scale, rotation and translation). Two points of correspondence are required to uniquely define such a transformation, termed a linear conformal affine transform (LCAT).

Given a set of 2D putative correspondences $\mathcal{P}^S \mapsto \mathcal{P}^I = \{p_{i=1...m}, p_{s=1...n}\}$, our iterative search runs as follows. We randomly sample two pairs of correspondences, deducing the LCAT (M). We then compute the forward-backward transfer error $E(T; \mathcal{P}^S_m, \mathcal{P}^I_n)$ using all putative correspondences:

$$E(M) = \sum_{\{p_s, p_i\} \in \mathcal{P}^S \mapsto \mathcal{P}^I} |p_s - M p_i|^2 + |p_i - M^{-1} p_s|^2 \quad (5)$$

We iterate for up to 10,000 trials seeking to minimize $E(M)$.

5. Incorporating Semantics

Sketch queries can concisely specify target appearance, yet the target semantics (*e.g.* butterfly, London) are often more concisely specified using text. We can therefore enhance our sketch based image retrieval and photo montage systems by incorporating semantic information, using any keyword (tag) meta-data accompanying images in the dataset (also harvested from Flickr).

5.1. Tag similarity

Word co-occurrence is one of the most commonly used methods to measure the similarity between tag pairs. In this paper, we adapt the co-occurrence based tag similarity graph with the shortest path algorithm, which is shown to improve the retrieval performance on our dataset.

Given a vocabulary $\mathcal{V} = \{w_1, ..., w_K\}$ of K keywords present within all image tags, the similarity of a pair of keyword tags is commonly defined using tag co-occurrence. In this work we adopt an asymmetric measure [42] indicating the probability of w_i appearing in a tag-set, given the presence of w_j:

$$P(w_i|w_j) = \frac{|w_i \cap w_j|}{|w_j|}. \quad (6)$$

We model the vocabulary as a directed graph $\{\mathcal{V}, \mathcal{E}\}$ where edge $\mathcal{E} = p(w_i|w_j)$, see Fig. 5 (left). The co-occurrence metric represents the probability of two words co-occurring explicitly.

However, in the case of Flickr, the vocabulary is unconstrained and different users may prefer different synonyms to describe the same object. For example, the words 'sea' and 'ocean' may not explicitly co-occur but each may co-occur with common tags (e.g. 'beach') which imply the two words 'sea' and 'ocean' have a strong connection with each other, as well as with 'beach'. Thus it is also necessary to consider the implicit (transitive) relationship between words. Note that although 'WordNet' [63] shows the semantic structures of words and has been used for image retrieval, in our case the words relation shown by WordNet could be too broad for a particular subset of the world scenario.

We address this using Dijkstra's algorithm to find the shortest path between two nodes in the graph with weights initially derived via eq. (6). We evaluate the probability of a sink node (tag) given a source node via the shortest path cost. Since path cost is a summation, yet successive transitions between tags (nodes) imply a joint probability, we encode edge weights as log-likelihoods. Figure 5 (right) illustrates how the tag similarity graph is updated by this process.

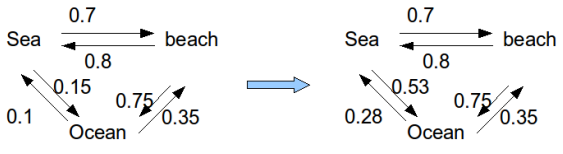

Figure 5: Tag similarity graph. Similarity based on co-occurrence metric (left) and the updated graph using the shortest path algorithm (right).

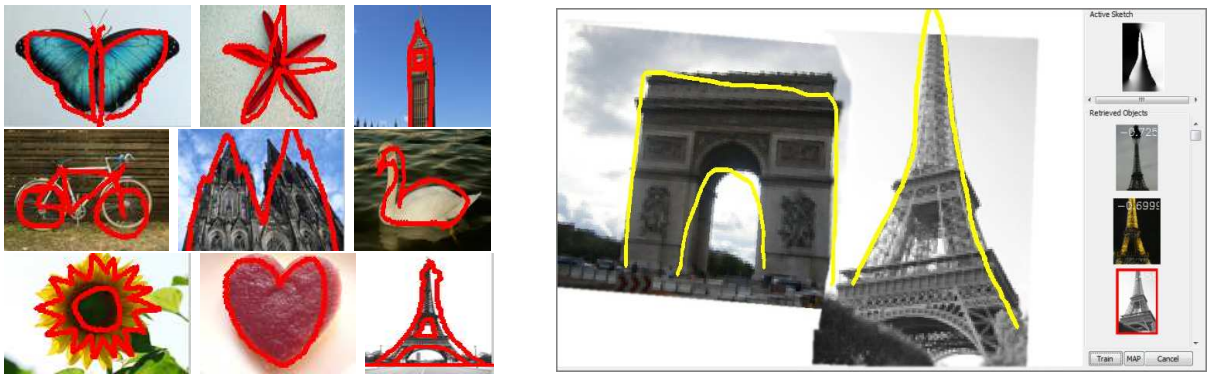

Figure 4: Sketch localization. Left: Localizing the query sketch within retrieved images using RANSAC/GF-HOG. Right: An interactive semantic photo montage application driven by our BoVW retrieval system. Sketches are combined with context from the tags of previously retrieved images, to suggest new elements for the montage. The sketch localization is harnessed to align the images for compositing at the appropriate position in the montage.

5.2. Semantic similarity of images

We now describe how to measure the image similarity in the semantic (tag) space. Multiple tags are typically assigned to an image and so we must consider the similarity of all pairings of tags when considering the semantic similarity of two images. We compute the similarity between two sets of tags $C^1 = C^1_1, C^1_2, ..., C^1_N$ and $C^2 = C^2_1, C^2_2, ..., C^2_M$, corresponding to images I^1 and I^2 as:

$$\frac{\sum_{m=1}^{M} \max_n\{p(C^1_n|C^2_m)\}}{N} + \frac{\sum_{n=1}^{N} \max_m\{p(C^1_n|C^2_m)\}}{M} \quad (7)$$

where $p(C^1_n|C^2_m)$ calculates the co-occurrence probability of two tags via the shortest path techniques of sub-Sec 5.1.

5.3. Combining sketch and text

We consider two different applications to integrate semantic information into our SBIR system: (1) combining keywords with a the free-hand sketch to create a hybrid retrieval system, accepting an *annotated sketch* as the input query; (2) a photo montage application wherein objects are sketched and results picked from a ranked list. Results are ranked according to similarity with the sketch, but also according to semantic similarity with retrieved objects already in the montage. The set of existing keyword meta-data adds context to the search.

5.3.1. Annotated sketches as queries

We incorporate semantic similarity into our interactive SBIR application, enabling users to input a set of keywords together with their hand drawn sketch. Fig. 13 illustrates the results of our hybrid approach, in which the semantic similarity (eq. 7) and the sketch similarity (table.1 and section 4.2) are combined via a simple weighted summation. Although our combination method is simplistic, the results show marked semantic improvement — for example comparing the results of the pyramid and circle sketches of Fig. 2. The search for a circle and London have returned images of the "London Eye", when our database contains many other circle shape images as well as other images of London not exhibiting circles. Similar observations are made of the triangular sketch and the Paris Louvre Museum.

5.3.2. Semantic photo montage

We demonstrate our retrieval and localization algorithms within a prototype system for photo montage using sketched queries (Fig. 4, right). Shapes are sketched, and selected from a retrieved list of results for compositing onto a canvas at the position sketched.

The system is similar in spirit to Chen *et al.*'s Sketch2Photo [39], except that we use sketched shape to retrieve our images rather than running a keyword search. Users sketch objects and select photos to insert into the composition from ranked results on the right. Upon selecting an image, the position of the sketch shape is localized and the region of interest cropped and composited into the sketch. Unlike Eitz *et al.*'s PhotoSketch [2] our GF-HOG enables matching invariant to scale and position of sketched shapes, and also combines the semantic keyword similarity score (eq. 7) to guide retrieval.

As the montage is constructed by retrieving and compositing objects, a set of relevant keywords is accumulated from the tags of images incorporated into the montage. This tag list provides a supporting context i.e. prior for subsequent sketch retrieval. The query sketch and keyword list are submitted for retrieval as

per Sec. 5.3.1 with the user able vary to balance their search requirements via interactively varying the weight between text and sketch. An example screen-shot of our photo montage system is shown in Fig. 4 (b).

6. Experiments

We present a comprehensive performance evaluation of GF-HOG within a BoVW framework, comparing against state of the art descriptors for shape and visual search across a variety of conditions (distance metrics and vocabulary sizes) critical to the performance of BoVW retrieval. We also investigate performance degradation over affine deformation of the sketched query, comparing to other state of the art descriptors. Run-time characteristics of the offline descriptor extraction, and the online retrieval and localisation processes are also discussed.

6.1. Dataset and queryset

We created a dataset of around 15k photographs (Flickr15k)[1] sampled from Flickr under Creative Commons license. We search Flickr images by using query tags that describe shapes (*e.g.* heart shape, and *et.al.*); landmarks (*e.g.* Louvre, big ben, and *et.al.*); semantic objects (*e.g.* horse, butterfly, and *et.al.*) and tags that describe a particular scene (*e.g.* beach, sun set, and *et.al.*). The first 1000 returned images for each query are downloaded together with their metadata (tags). We manually select images with a reasonable size of a meaningfull object area. This leaves us with 14660 images (denoted as the 'Flickr15K' dataset). The selected images exhibit significant affine variation in appearance and in presence of background clutter. Those images are then manually labelled into 33 categories based on shape only to evaluate our SBIR system. Around 10% of the set serves as noise i.e. not fitting any category. Each of the shape categories could contain multiple semantic objects. We further manually annotated five semantic categories considering both shape and semantics, according to the annotated sketch queries shown in the first column of Fig. 13 to evaluate our semantic SBIR system.

We recruited 10 non-expert sketchers to provide free-hand sketched queries of the objects. The participants comprised 4 undergraduate and 6 graduate volunteers (5 male, 5 female) recruited from outside our research group . All self-assessed their sketch skill level as average; this was the only criteria for candidate selection. In our study, participants were briefed with example images for each category, and the sketches are then drawn based on their memory. Participants worked individually without sight of each others work. In some cases this led to highly stylized representations of objects containing hallucinated detail not present in the images themselves (Fig. 6). In total, 330 sketches were generated i.e. one per category for each of the 10 participants[2].

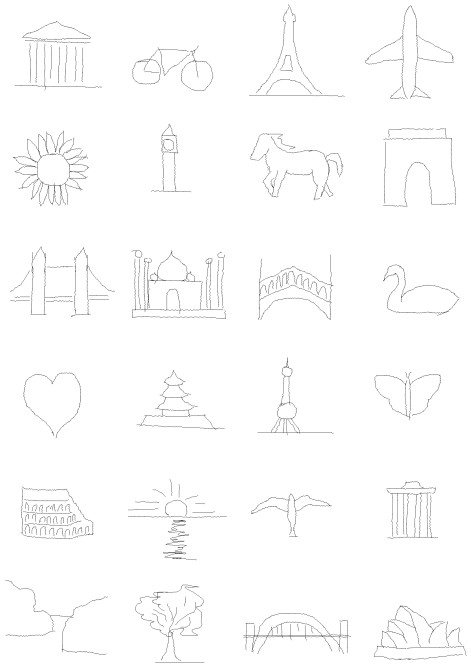

Figure 6: A representative sample of our 330 query sketch set used in evaluation, drawn by 10 non-expert participants.

This dataset (Flickr15k) is a novel large scale shape dataset of natural images, with the semantic tags, the labelled categories and the 330 free-hand sketch queries. It can be used as a benchmark for shape retrieval, classification and sketch based image retrieval.

6.2. Experimental setup

Whilst a variety of local descriptors such as SIFT, SSIM, HOG have been successfully used in image retrieval and classification tasks [57], it is still unclear how various local descriptors perform in SBIR. We now compare proposed GF-HOG with SIFT [62], Self Similarity (SSIM) [36], Shape Context [64], HOG [50] and

[1] We undertake to release this dataset on the public web, along with the source-code for our descriptor and the evaluation environment, post-review.

[2] The sketch query set will be publicly released post-review.

the Structure Tensor (used for SBIR in [2]). The parameters for GF-HOG were described in Subsec. 3.2. We now outline the parameters and setup used for the other descriptors below. In all experiments, the photos and the sketch canvas were pre-scaled so that their largest dimension (e.g. width or height) was 200 pixels. For each descriptor we compare against a publicly available, third-party authored reference implementation in C/C++.

6.2.1. SIFT

We compute the SIFT [62] descriptor treating each pixel of the edge map (for database images) or stroke mask (for sketches) as a key-point. SIFT is computed as described in [62, 65] within a 15×15 window local to the key-point. We used the University of Amsterdam (UoA) C/C++ (binary) benchmark implementation of SIFT [65], commonly adopted as a standard in international benchmark trials e.g. ImageCLEF, TrecVID.

6.2.2. SSIM

Shechtman and Irani [36] propose Self-similarity (SSIM) as an image descriptor invariant to depictive style. Recently, Chatfield *et al.* [66] report experiments using SSIM in a BoVW framework to retrieve photos using photo-realistic queries of shapes. SSIM is essentially an auto-correlation via sum-of-square-difference of a patch with its surrounding neighborhood. This correlation surface is then transformed into a binned log-polar representation. In our experiments, we compute the SSIM descriptor over the edge map (for database images) and sketched image. SSIM is computed using a 5×5 correlation window, over a larger 40×40 neighborhood. The SSIM correlation surface is partitioned into 36 log-polar bins (3 angles, 12 radial intervals). We used the defacto C/C++ reference implementation of SSIM supplied by the Oxford Visual Geometry Group (VGG) [67].

6.2.3. Shape Context

Shape Context [64] has been previously used to measure similarity, and establish correspondence, between shapes (including sketched shapes) for example [39] and so is evaluated here. A shape contour is represented as a set of discrete points, and each point p is described by the set of vectors originating from itself to all the other sample points on the shape. The histogram of p of these vectors is binned locally in log-polar space to yield the shape context. We use the C/C++ (Matlab mex) implementation supplied by the authors.

6.2.4. HOG (EDGE-HOG, S-HOG)

As discussed in subsec. 3.2.1 we compute the standard HOG descriptor over the same scale range as GF-HOG, sampling over all edge (or stroke) pixels. This practice was referred to as EDGE-HOG in [6] and S-HOG in [38]. In addition to using identical multiple scales, the same windowing parameters are also used. We use the C/C++ reference implementation of HOG supplied with the open source OpenCV library version 2.2, ported from the original code by Dalal and Triggs [50].

Table 2: Results of classical similarity measures

Descriptors	Distance measures	k	MAP
GF-HOG	Hist. I'sect	3500	0.1222
HOG	Chi^2	3000	0.1093
SIFT	Chi^2	1000	0.0911
SSIM	Chi^2	500	0.0957
ShapeContext	Chi^2	3500	0.0814
StructureTensor	Chi^2	500	0.0798

Table 3: Results of language model similarity measures

Descriptors	Language models	k	MAP
GF-HOG	Okapi	5000	0.08997
HOG	Okapi	10000	0.06654
SIFT	Okapi	6500	0.06244
SSIM	Okapi	2000	0.06257
ShapeContext	k-l divergence	500	0.04377
StructureTensor	k-l divergence	500	0.0559

6.2.5. Structure Tensor Descriptor

Eitz *et al.* [2] presented a grid based image matching using structure tensor descriptor. Images and sketches are decomposed into $a \times b$ grid of cells. The structure tensor $\mathcal{T}(I)$ is computed densely over greyscale image I and averaged within each cell to form a descriptor:

$$\mathcal{T}(I) = \begin{pmatrix} \frac{\delta I}{\delta x}^2 & \frac{\delta I}{\delta x}\frac{\delta I}{\delta y} \\ \frac{\delta I}{\delta x}\frac{\delta I}{\delta y} & \frac{\delta I}{\delta y}^2 \end{pmatrix} \quad (8)$$

The descriptors from each cell are concatenated to form an image descriptor. In our experiments we compare both to this gridded approach to structure tensor matching (using a 10×10 grid), and to the use of structure tensor as a local descriptor within a BoVW framework. In that latter case we compute the structure tensor local to edge or sketch pixels using a 15×15 window. The structure tensor was implemented using the C/C++ OpenCV function preCornerDetect to compute the three compound first derivatives of (8).

6.3. Comparative evaluation of GF-HOG

We perform BoVW retrieval using the proposed GF-HOG descriptor, implemented in C/C++, comparing

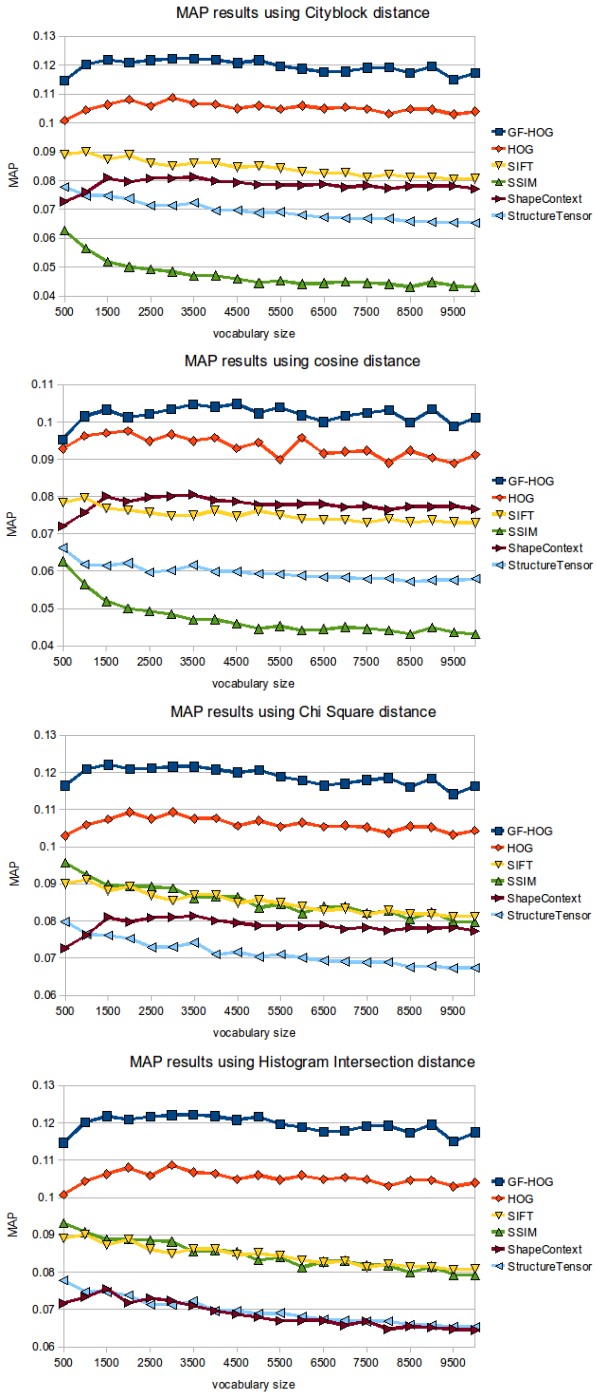

Figure 7: Performance (MAP) of our system vs. codebook size, comparing six descriptors over Flickr 15K dataset using four classical distance measures. From top to bottom: Cityblock distance; Cosine distance; χ^2 distance; Histogram intersection distance.

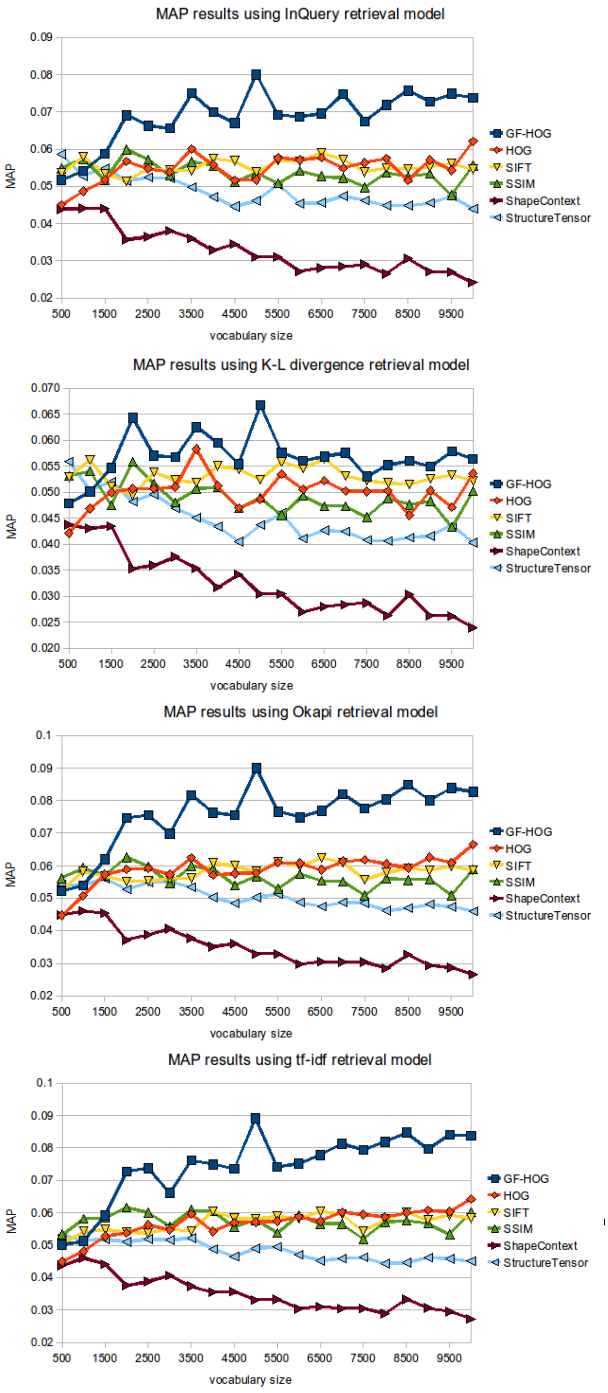

Figure 8: Performance (MAP) of our system vs. codebook size, comparing six descriptors over Flickr 15K dataset using four different language retrieval models. From top to bottom: InQuery; kl; Okapi; tf-idf.

performance with an otherwise identical baseline system incorporating our spectrum of alternative descriptors. Average Precision (AP) is computed for each query, and averaged over the query set to produce

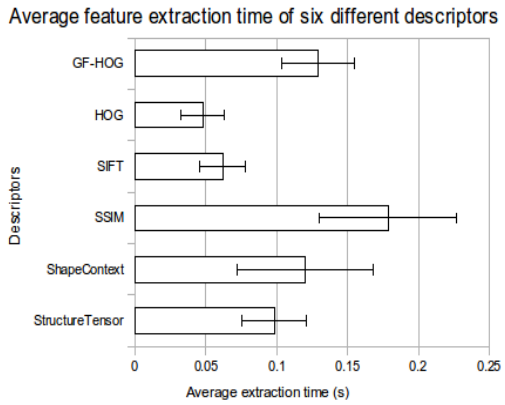

Figure 9: Average extraction time of six different features. Error bars indicate one standard deviation.

.

Mean Average Precision (MAP) score from the result rankings. MAP was computed using a widely used implementation for MAP scoring distributed via the TRECVID benchmark. Fig. 7 present these MAP scores over a range of vocabulary (codebook) size k, using a variety of distance measures commonly used in the Computer Vision and the Information Retrieval literature. As an additional non-BoVW baseline outside our framework, we computed MAP for our 330 sketch set using the recent SBIR system of Eitz et al. [2], yielding a MAP score of 7.35%.

Examining all distance measures, we observe the relative performances of the different descriptors stabilizes at vocabulary size of around $k = 3000$ for the "classical" distance measures and at around $k = 4000$ for the distance measures derived from text retrieval. The best performance is demonstrated using GF-HOG over the χ^2 and histogram intersection distances, with the MAP results differing only to the third decimal place due to performance deviation around $k = 6000 - 8000$. City-block (L^1) distance indicates around 1% lower performance than histogram intersection and χ^2 tests, which may prove a convenient trade-off in the implementation of structures such as $kd-$trees.

The trends of Figs. 7, 8 show for all distance measures a significant improvement in use of GF-HOG over contemporary descriptors, with regular HOG (EDGE/S-HOG) computed over multiple scales around 2% lower. The best performance settings observed for each descriptor are presented in Tables 2,3. Given a 33 category dataset, the random response for comparison is expected around 3% although as noted around 10% of the data has no category so actual random response is around 2.7%. The Structure Tensor descriptor, when encoded into a BoVW framework, performs poorly compared to the griding based strategy of [2] and appears unsuitable for implementation in BoVW above $k = 500$. This may be due to the relative low-dimensionality of the descriptor. Surprisingly the SSIM descriptor performs relatively poorly compared to other more classical descriptors (e.g. SIFT, HOG) not designed for depiction invariant matching. It may be that the retrieval success reported in a prior BoVW implementation of SSIM [66] was reinforced by either the post-process localization scheme based on Hough voting, or by the use of photographic rather than sketched queries. Although the relative performances of descriptors stays relatively constant across the classical distance measures (Fig. 7), the Shape Context appears to be more variable — though always out-performed by over 3% by gradient-orientation based descriptors (SIFT, HOG, GF-HOG). Our findings confirm the results in [6] and [38] in relation to higher performance of EDGE/S-HOG (i.e. HOG over edges) versus alternate descriptors, but indicate that gradient field interpolation prior to HOG delivers a significant benefit.

The language model based measures of distance are computed via Lemur, and due to their increased algorithmic complexity have produced more notably non-linear MAP responses from the BoVW system (Fig. 8). Nevertheless, across all three measures a clear ranking of descriptors can be observed from Shape Context performing most poorly (approaching random response as early as $k = 4000$) to GF-HOG out-performing others. However GF-HOG performance peaks at only around 7% MAP for these similarity measures, which appear inferior to classical distance measures for all descriptors.

Fig. 3 presents several queries from our evaluation set, and their results over Flickr15k. The results correspond closely to the sketched shape; despite BoVW abstracting away spatial relationships between features, the GF-HOG descriptor is able to capture this relative spatial information prior to coding. Although there are some explainable inaccuracies (e.g. between plane and starfish) the clear majority of results are relevant, including some interesting robust examples (mirrored images of the swan are returned).

6.4. Computational cost

Our experiments were performed on an Ubuntu 10.4 system with 2.2GHz Dual Core Intel Pentium 4 processor and 4 GB main memory supported by a 6144Kb cache. We measured computational cost in three key stages of our system: i) image feature extraction (preprocessing), ii) image retrieval at query-time, iii) localisation of the sketched object within an image.

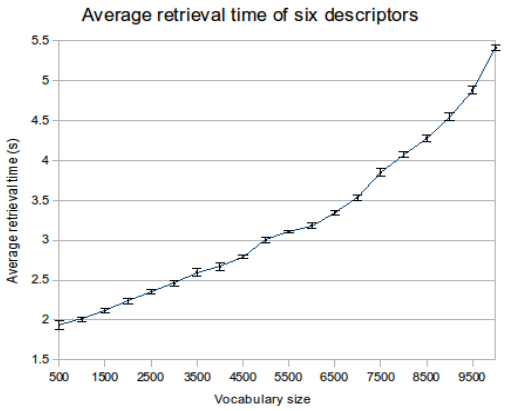

Figure 10: Average retrieval time of using six different descriptors with varying vocabulary size $k = [500, 10000]$ using histogram intersection over Flickr15k. Error bars indicate one standard deviation.

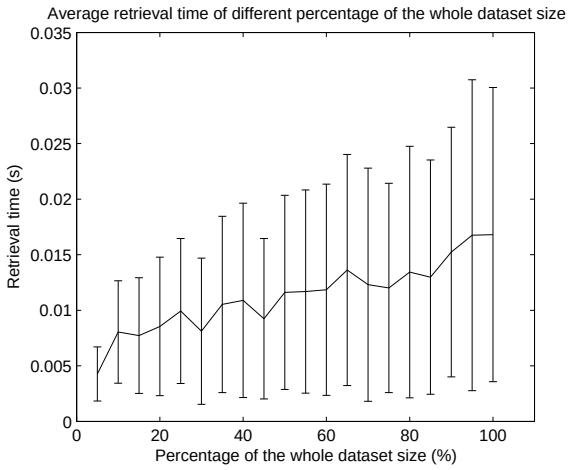

Figure 11: Average retrieval time of using different percentage of the whole dataset size with vocabulary size $k = 3500$ using Cityblock distance based kd-tree. Error bars indicate one standard deviation. Note the retrieval time shown in this graph only consider the matching time, does not include the feature extraction of the query sketch.

6.4.1. *Feature Extraction*

Fig. 9 indicates the time taken to extract descriptors from a single image, averaged over all images in the Flickr15k dataset. The experiment was repeated for each descriptor type. The resulting average times and standard deviations do not include the common first step (Canny edge extraction) which took on average 0.015s per image. We observe GF-HOG to compare favourably with ShapeContext and the much slower patch correlation based approach of Self-Similarity. The simple derivative (Structure Tensor) and gradient histogram (HoG, SIFT) based descriptors are faster to compute on average, but have been shown to yield lower accuracy. Unsurprisingly HoG is much quicker to compute than GF-HOG, since HoG is a sub-step within the GF-HOG process. On average, 41% of GF-HOG execution time is spent computing HoG, with the remainder used by the TAUCS library to solve the sparse linear system necessary to compute the gradient field.

6.4.2. *Image Retrieval*

As the BoVW image retrieval process is independent of the descriptor used, comparable timings are obtained for retrieval using each of the six descriptors. Retrieval times averaged over all descriptor types are shown in Fig. 10 for codebook sizes in range $k = [500, 10000]$, using the histogram intersection measure over the Flickr15k dataset. Timing invariance to descriptor type is reflected in the narrow error bars. Retrieval time scales approximately linearly with k. For best performing accuracy we observed $k = 3500$ (Table 2), for which retrieval time is on average 2.5s.

One advantage of our BoVW based SBIR system is that the retrieval efficiency can be easily improved by using techniques including hierarchical search, kd-tree, Locality-sensitive hashing, inverse document indexing. In this paper, we explore using the Cityblock distance based kd-tree to improve the retrieval time, since the Cityblock distance achieves comparable performance to the best results achieved by the Histogram Intersection distance (shown in Fig. 7) and its linear geometry nature makes it easy to be adapted in the kd-tree indexing technique. A graph of the average retrieval time with a varying percentage of the dataset is shown in Fig. 11, from which we can see the retrieval time increases sub-linearly with the size of the dataset. In our experiments of using Cityblock distance based linear search the retrieval time increases approximately linearly with the increasing database size. The total retrieval time when searching the entire dataset using Cityblock distance based linear search is 0.88s. This shows the effectiveness of using kd-tree to improve the efficiency of retrieval comparing with figure 11.

6.4.3. *Object Localisation*

The time taken to localise the sketched object within a selected image is non-deterministic due to the RANSAC process. The average time of localizing a single query sketch within a single returned image, averages at 0.267s with standard deviation 0.027. The average time spent to localize 50 query sketches within the corresponding top retrieved result is 0.263s with standard deviation 0.039.

6.5. *Affine invariance evaluation*

In order to evaluate the robustness of the descriptors we generated queries for nine translated, scaled

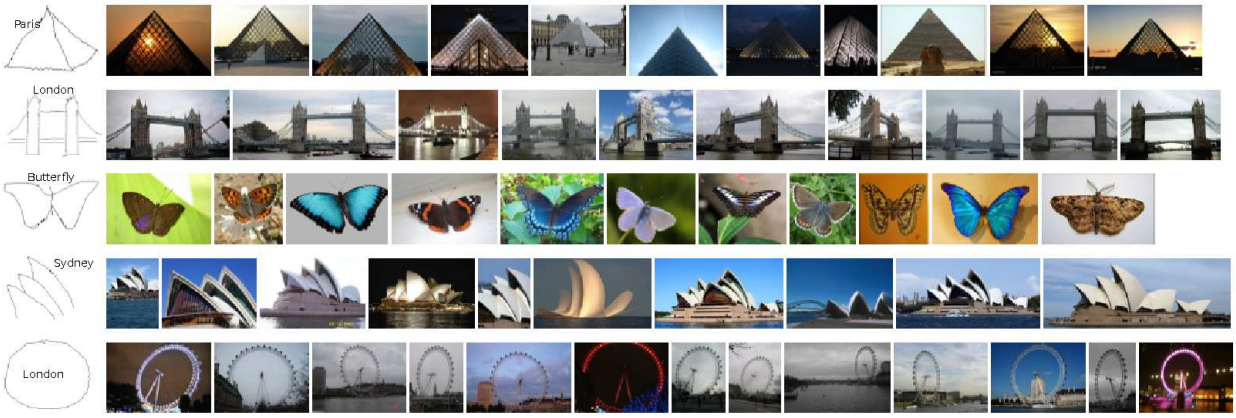

Figure 13: Annotated sketches as queries. Some hand picked results from the top 15 returned results by using both sketch and text modalities — compare the circle and triangle sketch to their equivalent sketch only searches in Fig. 3. The text provides additional semantic context that supplements the shape description in the sketch.

and rotated version of each input sketch. The translated sketches are generated by translating the original sketch by a random shift between [-0.8,0.8] multiples of the width of the sketch canvas, and also in a random direction. The scaled sketches are generated by scaling the input sketch uniformly by a random factor in range [0.6,1.4]. The rotated sketches are random rotations (about the centroid of stroke pixels in the sketch) in range [-20,20] degrees. In total we generated 24 affine transformed version for each of the 330 query sketches i.e. 7920 sketches in our query set.

Fig. 12 (a,b) shows the MAP results of the rotated and scaled versions of the query sketches. As expected, the greater the affine deviation of the sketch from the typical configuration of the target objects in each category, the greater the performance (MAP) degradation for the rotation and scaling. As the BoVW system abstracts away translation, all configurations of our BoVW system perform near-identically regardless of translation, making the system naturally translation invariant. We have therefore plotted in Fig. 12 only the response for our non-BoVW baseline (structure tensor grid)[2]. In that case, a translation of approximately 20% is sufficient to drop MAP by 1% as content moves into neighboring grid cells.

Based on our earlier observations regarding the MAP performance of BoVW systems peaking around $k = 3000$, using histogram intersection distance, these conditions were selected for use in the affine degradation experiment. Regarding rotation, Fig. 12 (a) shows that the MAP score of using any of the six descriptors degrades when the query sketch is rotated, but the rate of degradation for GF-HOG and the other gradient based descriptors is approximately equal. However the degradation for Structure Tensor (both with and without BoVW) and for Shape Context is more pronounced. Regarding scaling, enlarging of the sketch relative to its original size produces greater degradation than size reduction in all cases. However the degradation of GF-HOG is around twice as slow (loss of 0.5% per 10% scale increase) versus SIFT and edge based HOG (1% per 10% scale increase). Although some other descriptors degrade more gracefully, their overall MAP performance is less than half that of GF-HOG.

6.6. Annotated sketch queries for image retrieval

Fig. 13 shows examples of fusing sketches and text as search queries. Although in some cases a sketch alone is ambiguous (e.g. a circle) we can constrain the semantics of the results by adding just one or two tags as additional information. For example, compare the circle and the pyramid queries tagged here with “London” and “Paris” to the circle and pyramid sketches in Fig. 3 with no semantic constraint. The relevant images (e.g. London Eye are returned. Our dataset contains several examples of non-circular tagged London (e.g. tower bridge, Big Ben) and non-pyramidal images tagged Paris (including eiffeltower, the monument, and some building architecture). In our system, we show that a simple sketch together with a few ambiguous tags could help users picture the object in their mind and help them find the related results. This is particularly helpful when users only have an abstract shape in their mind with not detailed semantic information of an object. Many times neither of the shape nor the limited semantic information alone is sufficient, while the combination or these two can help search for the particular object.

Quantitative evaluation is also conducted to compare our shortest path based tag similarity with the stan-

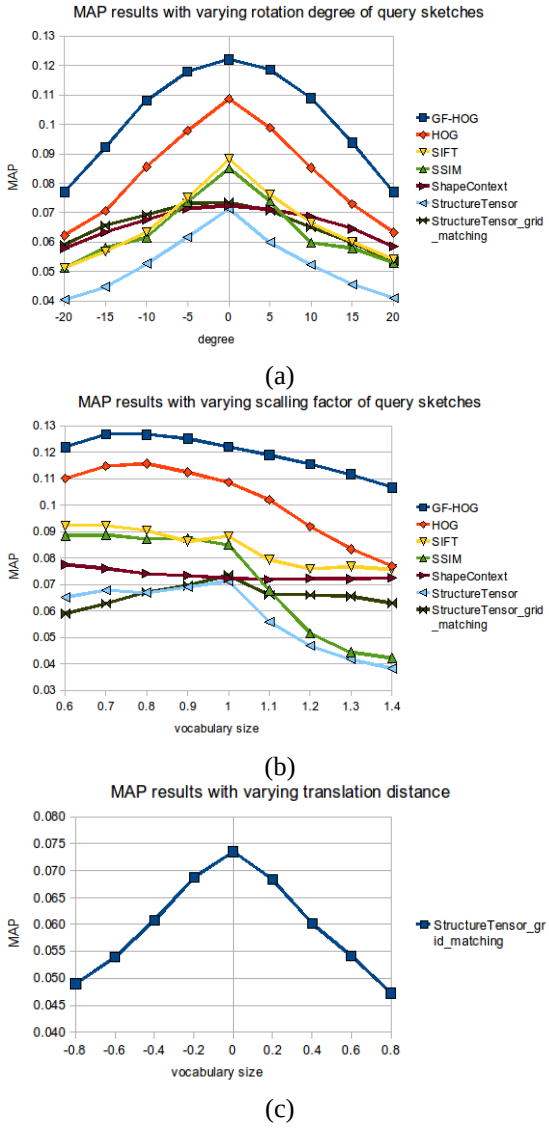

Figure 12: Evaluating performance degradation over affine warp of the query sketches, by (a) rotation; (b) scaling; (c) translation.

dard words coocurrence based tag similarity. Average precision-recall curves of the five annotated sketch queries as shown in the first column in Fig. 13 are shown in Fig. 14. From the curves we can see that our proposed shortest path based tag similarity measure achieves better performance on the top returned results. The precision drops quickly after recall reaches around 25%, after which the two curves share a similar shape. This may due to the poor quality of tags available for Flickr images. However, for a large scale image retrieval system users are more interested to the top returned results.

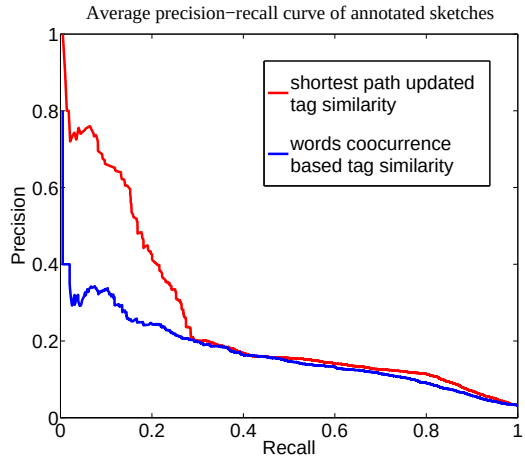

Figure 14: Average precision-recall curves of the five annotated sketch queries shown in the first column of Fig. 13.

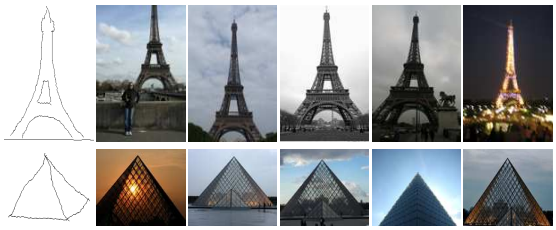

Figure 15: Searching in context. A sketch of the characteristically shaped Eiffel tower returns the expected result. The tags of the result image are used as context in a subsequent sketch search for a pyramid. Pyramidal images that share tags with the Eiffel Tower image (e.g. “Paris”) are ranked higher.

6.7. Semantic context for retrieval and photo montage

Instead of using annotated sketch queries, section 5.3.2 represented our system to incorporate the semantic information from search ontext. As demonstrated in our photo montage system (Fig. 4b) the context supplied by the tag set can also be drawn from the tags of other images. A further example of a search considring the semantic context is given in Fig. 15 where tags from the previously retrieved images of the Eiffel tower are used to constraint its successive search for triangular shapes, resulting in an image of the Louvre pyramid both tagged “Paris”.

7. Conclusion

We have described GF-HOG an image descriptor suitable for Sketch based Image Retrieval (SBIR) in a Bag of Visual Words (BoVW) framework. We have undertaken a comprehensive performance evaluation of GF-HOG under BoVW, comparing against several state

of the art descriptors (SIFT, SSIM, Shape Context, HOG, Structure Tensor) under a variety of conditions (vocabulary size, distance measure, affine variation). In all conditions we have demonstrated a superior MAP performance of GF-HOG for the task of SBIR over our Flickr15k dataset, averaged over 330 query sketches drawn from memory by 10 non-expert users. Detailed timing analysis has been conducted at all stages of the retrieval pipeline; feature extraction, image comparison, and localisation. These experiments demonstrate the improved accuracy of GF-HOG over other state of the descriptors art across a multitude of distance measures and affine variations. These results contribute to the SBIR community through aiding the selection of appropriate local descriptors for sketch matching, and the comparison of descriptor performance characteristics. Publication of our *descriptor source code* and the manually annotated Flickr15k dataset forms a further contribution to the SBIR community.

A photo montage system is implemented which use GF-HOG to match, and enable the sketch query to be localized within the retrieved images using a RANSAC process. We use a four-parameter affine transform, the matching results are used to create a prototype photo montage application in which sketched objects will be composited onto a canvas.

We have also demonstrated how appearance (sketched queries) search may be enhanced by semantics (metadata tags) to constraint searches to particular semantic objects. An adapted tag similarity using shortest path is proposed, which has shown improved performance than the standard tag co-occurrence based similarity measure. Although our method of combining text and sketch is simplistic, it is sufficient to show the power of combining text with sketch via our GF-HOG framework. Future directions of this work will explore more sophisticated combination schemes, for example kernel canonical correlation analysis (KCCA) [68] which has been used to good effect combining photo-realistic and textual constraints outside the domain of SBIR.

Although GF-HOG computed over Canny edge maps has here been shown to outperform state of the art descriptors for SBIR, research questions remain around the selection of scale for first deriving the Canny edge map. It may be possible to draw upon work on grouping regions for structure invariant matching [69], to select an appropriate set of scales for edge detection and further improve retrieval accuracy. However we believe such enhancements are not necessary to demonstrate the robustness and performance of GF-HOG for SBIR, and its potential for use in sketch based retrieval applications such as sketch-text search and photo montage.

References

[1] Bimbo, A. D., Pala, P., 1997. Visual image retrieval by elastic matching of user sketches. in: IEEE PAMI, pp. 121-132.

[2] Eitz, M., Hildebrand, K., Boubekeur, T., Alexa, M., 2009. A descriptor for large scale image retrieval based on sketched feature lines. In: SBIM. pp. 29-38.

[3] Tahir, M. A., Yan, F., Barnard, M., Awais, M., Mikolajczyk, K., Kittler, J., 2010. The university of surrey visual concept detection system at imageclef 2010: Working notes. In: ICPR.

[4] Sivic, J., Zisserman, A., 2003. Video Google: A text retrieval approach to object matching in videos. In: ICCV. Vol. 2. pp. 1470-1477.

[5] Chum, O., Philbin, J., Sivic, J., Isard, M., Zisserman, A., 2007. Total recall: Automatic query expansion with a generative feature model for object retrieval. In: ICCV.

[6] Hu, R., Barnard, M., Collomosse, J. P., 2010. Gradient field descriptor for sketch based retrieval and localization. In: ICIP. pp. 1025-1028.

[7] Kato, T., Kurita, T., Otsu, N., Hirata, K., 1992. A sketch retrieval method for full color image database query by visual example. In: ICPR. pp. 530-533.

[8] Shih, J.-L., Chen, L.-H., 2001. A new system for trademark segmentation and retrieval. in: Image Vision Computing, pp. 1011-1018.

[9] Leung, W. H., Chen, T., 2002. Trademark retrieval using contour-skeleton stroke classification, in: ICME, pp. 517-520.

[10] Fonseca, M. J., Ferreira, A., Jorge, J. A., 2005. Content-based retrieval of technical drawings, in: IJCAT. pp. 86-100.

[11] Liang, S. and Sun, Z., 2008. Sketch retrieval and relevance feedback with biased SVM classification, in: Pattern Recogn. Lett., pp. 1733-1741.

[12] Fonseca, M. J., Gonçalves, D., 2010. Sketch-a-Doc: Using Sketches to Find Documents, in: Proceedings of the 16th International Conference on Distributed Multimedia Systems (DMS), pp. 327-330.

[13] Fränti, P., Mednonogov, A., Kyrki, V., Kälviäinen, H., 2000. Content-based matching of line-drawing images using the Hough transform, in: International Journal on Document Analysis and Recognition, pp.24-5.

[14] Sousa, P., Fonseca, M. J., 2010. Sketch-Based Retrieval of Drawings using Spatial Proximity , in: Journal of Visual Languages and Computing (JVLC) pp.69-80.

[15] Wang, C., Zhang, J., Yang, B., Zhang, L., 2011. Sketch2Cartoon: composing cartoon images by sketching, in: ACM Multimedia, pp. 789-790.

[16] Liang, S., Sun, Z., Li, B., 2005, Sketch Retrieval Based on Spatial Relations, in: CGIV, pp. 24-29.

[17] Leung, W. H., Chen, T., 2002. Retrieval of sketches based on spatial relation between strokes, in: ICIP pp. 908-911.

[18] Fonseca, M. J., Ferreira, A., Jorge, J. A., 2009. Sketch-based retrieval of complex drawings using hierarchical topology and geometry, in: Comput. Aided Des., pp. 1067-1081.

[19] Qiu, G., 2002. Indexing chromatic and achromatic patterns for content-based colour image retrieval. in: Pattern Recognition, pp. 1675-1686.

[20] Ashley, J., Flickner, M., Hafner, J. L., Lee, D., Niblack, W., Petkovic, D., 1995. The query by image content (QBIC) system. In: SIGMOD Conference. p. 475.

[21] Smith, J., Chang, S.-F., 1996. Visualseek: a fully automated content-based image query system. In: ACM Multimedia. pp. 87-98.

[22] Jacobs, C. E., Finkelstein, A., Salesin, D. H., Aug. 1995. Fast multi-resolution image querying. In: Proc. ACM SIGGRAPH. pp. 277-286.

[23] Do, M. N., Vetterli, M., 2002. Wavelet-based texture retrieval using generalized gaussian density and kullback-leibler distance. in: IEEE Trans. Image Processing, pp. 146-158.

[24] Pi, M. H., Tong, C. S., Choy, S. K., Zhang, H., 2006. A fast and effective model for wavelet subband histograms and its application in texture image retrieval.
[25] Choy, S.-K., Tong, C.-S., 2010. Statistical wavelet subband characterization based on generalized gamma density and its application in texture retrieval. in: IEEE Trans. on Image Processing 19 (2), pp. 281-289.
[26] Sciascio, E., Mongiello, M., Mongiello, M., 1999. Content-based image retrieval over the web using query by sketch and relevance feedback. In: Intl. Conf. on Visual Information Systems. pp. 123-130.
[27] Matusiak, S., Daoudi, M., Blu, T., Avaro, O., 1998. Sketch-based images database retrieval. In: International Workshop on Advances in Multimedia Information Systems. pp. 185-191.
[28] Mokhtarian, F., Mackworth, A. K., August 1992. A theory of multiscale, curvature-based shape representation for planar curves. in: PAMI, pp. 789-805.
[29] del Bimbo, A., Pala, P., 1997. Visual image retrieval by elastic matching of user sketches, in: PAMI, pp. 121-132.
[30] Ip, H. H. S., Cheng, A. K. Y., Wong, W. Y. F., Feng, J., 2001. Affine-invariant sketch-based retrieval of images. In: International Conference on Computer Graphics. pp. 55-61.
[31] Jain, A. K., Vailaya, A., 1996. Image retrieval using color and shape. in: Pattern Recognition, pp. 1233-1244.
[32] Chans, Y., Lei, Z., Lopresti, D., Kung, S. Y., 1997. A feature-based approach for image retrieval by sketch. In: SPIE Storage and retrieval for image and video databases.
[33] Rajendran, R., Chang, S., 2000. Image retrieval with sketches and compositions. In: ICME. pp. 717-720.
[34] Chalechale, A., Naghdy, G., Mertins, A., 2005. Sketch-based image matching using angular partitioning. in: IEEE Trans. Systems, Man, and Cybernetics, pp. 28-41.
[35] Tao, L., Yuan, L., Sun, J., 2009. Skyfinder: Attribute-based sky image search. In: ACM Trans. Graph.
[36] Shechtman, E., Irani, M., June 2007. Matching local self-similarities across images and videos. In: CVPR.
[37] Wang, C., Cao, Y., Zhang, L., 2011. Mindfinder: A sketch based image search engine based on edge index. In: Proc. Comp. Vision and Pattern Recognition.
[38] Eitz, M., Hildebrand, K., Boubekeur, T., Alexa, M., 2010. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. In: TVCG. Vol. 99.
[39] Chen, T., Cheng, M.-M., Tan, P., Ariel, S., Hu, S.-M., 2009. Sketch2Photo: internet image montage. in: ACM Trans. Graph. 28 (5), pp. 1-10.
[40] Wang, C., Li, Z., Zhang, L., 2010. Mindfinder: image search by interactive sketching and tagging. In: WWW. pp. 1309-1312.
[41] Cao, Y., Wang, H., Wang, C., Li, Z., Zhang, L., Zhang, L., 2010. Mindfinder: interactive sketch-based image search on millions of images. In: ACM Multimedia. pp. 1605-1608.
[42] Sigurbjörnsson, B., van Zwol, R., 2008. Flickr tag recommendation based on collective knowledge. In: WWW. pp. 327-336.
[43] Wartena, C., Brussee, R., Wibbels, M., 2009. Using tag co-occurrence for recommendation. In: 19th International Conference on Intelligent Systems Design and Applications. pp. 273-278.
[44] Schmitz, P., 2006. Inducing ontology from flickr tags. In: Proceedings of the Collaborative Web Tagging Workshop.
[45] Escalante, H. J., Montes, M., Sucar, L. E., 2008. Improving automatic image annotation based on word co-occurrence, pp. 57-70.
[46] Isaac, A., van der Meij, L., Schlobach, S., Wang, S., 2007. An empirical study of instance-based ontology matching. In: ISWC/ASWC. pp. 253-266.
[47] Agrawal, R., Imielinski, T., Swami, A., 1993. Mining association rules between sets of items in large databases. pp. 207-216.
[48] Lipczak, M., 2008. Tag recommendation for folksonomies oriented towards individual users. In: ECML/PKDD Discovery Challenge Workshop.
[49] Hu, R., James, S., Collomosse, J. P., 2012. Annotated Free-Hand Sketches for Video Retrieval Using Object Semantics and Motion, in: MMM, pp. 473-484.
[50] Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. In: CVPR. pp. 886-893.
[51] Pérez, P., Gangnet, M., Blake, A., 2003. Poisson image editing. in: ACM Trans. Graph. 22 (3), pp. 313-318.
[52] Toldeo, S., Chen, D., Rotkin, V. TAUCS: A library of sparse linear solvers. http://www.tau.ac.il/ stoledo/taucs/.
[53] Bosch, A., Zisserman, A., Munoz, X., 2007. Representing shape with a spatial pyramid kernel. In: CIVR. pp. 401-408.
[54] Yang, J., Jiang, Y.-G., Hauptmann, A. G., Ngo, C.-W., 2007. Evaluating bag-of-visual-words representations in scene classification. In: MIR. pp. 197-206.
[55] Geng, B., Yang, L., Xu, C., 2009. A study of language model for image retrieval. In: Proceedings of the 2009 IEEE International Conference on Data Mining Workshops. pp. 158-163.
[56] Mikolajczyk, K., Schmid, C., October 2004. Scale & affine invariant interest point detectors. in: IJCV, pp.63-86.
[57] Mikolajczyk, K., Schmid, C., 2005. A performance evaluation of local descriptors. in: PAMI 27 (10), pp.1615-1630.
[58] Mikolajczyk, K., Uemura, H., 2011. Action recognition with appearance-motion features and fast search trees. in: Computer Vision and Image Understanding 115 (3), pp. 426-438.
[59] Ren, R., Collomosse, J. P., Jose, J. M., 2011. A BoVW based query generative model. In: MMM. pp. 118-128.
[60] The Lemur project. `http://www.lemurproject.org`. Accessed March 2011.
[61] Callan, J., Lu, Z., Croft, W., 1995. Searching distributed collections with inference networks. In: Proc. ACM SIGIR. pp. 21-29.
[62] Lowe, D., 2004. Distinctive image features from scale-invariant keypoints. IJCV 60, 91-110.
[63] Fellbaum, C., 1998. WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.
[64] Belongie, S., Malik, J., Puzicha, J., 2002. Shape matching and object recognition using shape contexts. in: PAMI, PP. 509-522.
[65] van de Sande, K. E. A., Gevers, T., Snoek, C. G. M., 2010. Evaluating color descriptors for object and scene recognition. in: PAMI, pp. 1582-1596.
[66] Chatfield, K., Philbin, J., Zisserman, A., 2009. Efficient retrieval of deformable shape classes using local self-similarities. in: ICCV Workshops, pp. 264-271.
[67] Gulshan, V., Visual geometry group: Self-simi4larity descriptor. http://www.robots.ox.ac.uk/~vgg/software/SelfSimilarity, last accessed, June 2011.
[68] Hwang, S. J., Grauman, K., 2010. Accounting for the relative importance of objects in image retrieval. In: BMVC. pp. 58.1-58.12.
[69] Bai, X., Song, Y.-Z., Baliki, A., Hall, P., 2008. Structure is a visual class invariant. In: Proc. SSPR and SPR (Lec. Notes in Comp. Sci).