# Semantic Retrieval and Automatic Annotation: Linear Transformations, Correlation and Semantic Spaces

Jonathon S. Hare and Paul H. Lewis

School of Electronics and Computer Science, University of Southampton, Southampton, SO17 1BJ, UK

## ABSTRACT

This paper proposes a new technique for auto-annotation and semantic retrieval based upon the idea of linearly mapping an image feature space to a keyword space. The new technique is compared to several related techniques, and a number of salient points about each of the techniques are discussed and contrasted. The paper also discusses how these techniques might actually scale to a real-world retrieval problem, and demonstrates this though a case study of a semantic retrieval technique being used on a real-world data-set (with a mix of annotated and unannotated images) from a picture library.

**Keywords:** Semantic Image Retrieval, Automatic Annotation, Visual-terms, Evaluation, Correlation, Linear Transformation, Semantic spaces, Latent Semantic Analysis

## 1. INTRODUCTION

"I need some pictures of agricultural scenes, but preferably without any machinery or horses in it. It needs to be *timeless*." The previous quotation is of a real request to a picture librarian at a large archive. The query is typical of the kinds of things professionals search images for. If an image corpus is fully indexed using some or all of the words in the query, then the retrieval problem can be solved using existing text indexing and understanding techniques. Unfortunately, providing this kind of annotation and indexing for large collections of images is a slow and expensive task when performed manually.

Current research on automatic annotation and semantic retrieval aims to work towards finding a solution to the problem of automatically indexing unannotated image collections with the aim of making the corpus of images as accessible to retrieval and semantic understanding as a text corpus is now. The problem of how to get from the raw pixel content of images to semantic meaning has become known as the problem of the *semantic gap* in image retrieval.[1–4]

Techniques for attempting to bridge the semantic gap in image retrieval have mostly used an *auto-annotation* approach, in which keyword annotations are applied to unlabelled images (e.g. Ref. 5–8). The basic premise of these automatic annotation approaches is that a model can be learnt from a training set of images that describes how low-level image features are related to higher-level keywords. This model can then be applied to unannotated images in order to automatically generate keywords that describe their content. In essence, the process of auto-annotation is analogous to translating from one language to another.[7,8] In fact, many of the state-of-the-art techniques for encoding low-level image content are based around the idea of transforming or quantising the features to a vocabulary of visual terms, which represent a purely visual language.[9,10] A recent review of a number of automatic annotation and semantic retrieval techniques can be found in Ref. 11.

One of the problems with current auto-annotation approaches with regard to multimedia retrieval is that they can seriously harm retrieval effectiveness if the annotations they provide are wrong. This problem is partially addressed with relatively recent probabilistic auto-annotation[6,12] and our own semantic space[13] approaches to retrieval, which do not actually assign keywords to multimedia documents, but instead rank them by their likely similarity to a textual query. Fundamentally, a semantic space is a large multidimensional space in which objects are placed. In terms of image retrieval, these objects fall into three classes: keywords, visual-terms, and images.

Further author information E-mail: jsh2@ecs.soton.ac.uk

The placement of these objects into the space is such that items that are semantically related will appear in similar spatial locations.[13]

This paper introduces a new technique for automatic annotation and semantic retrieval that is based on the idea of deriving a transformation from the visual-term space formed by an image directly to the keyword space formed by annotations. The paper then describes and illustrates how this technique performs with reference to other pre-existing but related techniques. The advantages and disadvantages of each of the techniques are discussed. A discussion of the application of one of the semantic retrieval techniques to a real image collection is also included.

## 2. VECTOR-SPACES FOR SEMANTIC RETRIEVAL

As mentioned, it has become popular to transform image features into discrete elements or *terms*. These so-called "visual terms" are elegant because they enable image content to be described in much the same way as a text document. Techniques for creating visual terms from features almost always revolve around the idea of quantising the features into a fixed number of discrete values. At the simplest level, this can mean creating a visual term of each pixel in an image by quantising its colour value to say one of sixty-four allowed values. At the other end of the spectrum, visual terms may be created from SIFT features[14] created from the pixel content of salient regions, by quantizing them using a vocabulary learned by clustering an exemplar set of features (e.g. Ref. 9,10), or by segmenting an image into multiple segments and quantising these segments to a discrete vocabulary, again learnt through some form of clustering (e.g. Ref. 7, 8, 15).

The problem of automatically annotating images with keywords has often been approached from the problem of building a machine capable of translating from visual-terms to keyword annotation-terms. In particular, a *bag-of-words* model is often used in which the occurrences of terms in a given image are represented by term-occurrence vectors or histograms. The use of a bag-of-words model has one small limitation in that it implies that any image containing the same semantics and visual features is equivalent; that is to say the position of visual elements in an image is not important. In the following subsections we describe a number of relatively simple techniques for building such a machine. All of these techniques in their current form assume that their internal models are learnt in a batch mode from a large set of training examples that include occurrence vectors from both the visual-terms and keyword terms.

The first technique proposes a novel linear-transformation based approach to the problem in which an optimal direct mapping between the space spanned by the visual-term occurrence vectors and the space spanned by the (key)word occurrence vectors is learnt. The second set of techniques described were proposed by Pan *et al*;[15] the techniques work by learning a translation table that gives the probability of a word given a visual-term. These techniques are very similar to the first technique in that a direct mapping from visual-term space to word space is deduced. In order to perform semantic retrieval with all these techniques, images can be ranked based on the (decreasing) magnitude of the element of predicted words vector corresponding to a given query word.

The final technique discussed is the linear algebraic semantic space proposed by Hare *et al*.[13] This technique produces a vector space into which both visual-terms and keyword terms are mapped along with the images. Unannotated images can then be projected into this space. Semantic retrieval can be performed directly in this space by determining the location of a query word and ranking images based on their increasing distance from the query coordinates within the space.

### 2.1 Singular Value Decomposition

All of the techniques described in this section make use of a mathematical factorisation called the Singular Value Decomposition (SVD). Briefly, SVD is used to decompose matrix $\mathbf{A}$ into the product of three separate matrices, $\mathbf{U}$, $\mathbf{\Sigma}$, $\mathbf{V}^{\mathrm{T}}$:

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}} \tag{1}$$

The monotonically decreasing (in value) diagonal elements of the matrix $\mathbf{\Sigma}$ are called the singular values of the matrix A. These matrices represent the breakdown of the original relationships into linearly-independent vectors or factor values. By selecting the first (largest) $k$ singular values of $\mathbf{A}$, it is possible to construct a rank-$k$ approximation to $\mathbf{A}$ via $\mathbf{A}_k = \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^{\mathrm{T}}$ . A theorem by Eckart and Young[16] suggests that the $\mathbf{A}_k$ constructed

from the largest $k$ singular values of $\mathbf{A}$ is the closest rank-k approximation (in the least squares sense) to $\mathbf{A}$. By reducing the dimensionality of $\mathbf{A}$, it is possible to reduce the amount of "noise" in the original matrix. This is especially useful when the input matrix is formed from real world data, such as image features,[13,15,17] or the counts of words from text documents.[18]

## 2.2 Linear-transformation

In this section we propose a simple model that is capable of learning the translations or transformations of vectors of visual term occurrences to vectors of word occurrences. We formulate the problem as thus:

Assume $\mathbf{F}$ is an $n_f \times m$ matrix of $m$ training images represented by $n_f$ visual terms. Each element $f_{i,j}$ of $\mathbf{F}$ represents the number of times visual-term $i$ occurs in image $j$. Similarly assume $\mathbf{W}$ is an $n_w \times m$ matrix of the same $m$ training images represented by $n_w$ annotation words, and that the element $w_{i,j}$ represents the number of occurrences of word $i$ in image $j$. With most current image data-sets $\mathbf{W}$ is most probably binary in nature. Now assume that there exists a purely linear mapping between $\mathbf{F}$ and $\mathbf{W}$. This in itself is not an unreasonable assumption; we are only mandating that a linear combination of visual terms maps a given linear combinations of annotation words. This can be formulated mathematically as:

$$\mathbf{FT} = \mathbf{W} \tag{2}$$

If we can solve Equation 2 for $\mathbf{T}$, we then have a way of estimating the vector of words, $\vec{w}$ belonging to an unannotated document $\vec{u}$ by calculating:

$$\vec{u}\mathbf{T} = \vec{w} \tag{3}$$

Given the nature of the data, in all likelihood the system is over-determined, so there exist many possible solutions to $\mathbf{T}$, however we can select a solution that is in some sense optimal. A common approach is to choose to minimise the Euclidean norm, $||\mathbf{FT} - \mathbf{W}||^2$, using the Moore-Penrose pseudoinverse, $\mathbf{F}^+$:

$$\mathbf{T} = \mathbf{F}^+\mathbf{W} \tag{4}$$

The pseudoinverse $\mathbf{F}^+$ can easily be calculated using the Singular Value Decomposition $\mathbf{F} = \mathbf{U\Sigma V}^{\mathrm{T}}$ of $\mathbf{F}$, such that $\mathbf{F}^+ = \mathbf{V\Sigma}^+\mathbf{U}^{\mathrm{T}}$ where $\mathbf{\Sigma}^+$ is formed by simply replacing the non-zero elements of the diagonal matrix $\mathbf{\Sigma}$ with their reciprocal.

### 2.2.1 Noise reduction

The use of SVD in calculating the pseudoinverse gives an additional advantage in that the dimensionality of the $\mathbf{F}$ matrix can be reduced by setting small singular values in the $\mathbf{\Sigma}$ matrix to zero as described earlier. Previous work on content-based image retrieval has shown that reducing the dimensionality of a feature space can be advantageous in terms of promoting better semantic similarity of images.[10]

### 2.2.2 Weighting

Often, it can be prudent to weight the term-occurrence data matrices to reduce the effect of terms that occur commonly in lots of documents. Such terms tend not to have much discriminative power. Terms that occur rarely in only a few documents (and are thus more "unique") are obviously more discriminatory and useful in the analysis. One way to weight a matrix of term-occurrences is as follows; If we let $z_j$ be the number of images that contain the term $t_i$, the $i,j$-th element, $e_{i,j}$ of a term-occurence matrix can be weighted as:

$$e_{weighted_{i,j}} = e_{i,j} \times \log(\frac{N}{z_j}) \tag{5}$$

where $N$ is the total number of images. The weighting could be applied to either or both of $\mathbf{W}$ or $\mathbf{F}$. In text-retrieval the factor $\log(\frac{N}{z_j})$ is commonly used and is known as the Inverse Document Frequency (IDF).

### 2.3 Correlation and Similarity

Pan *et al.*[15] described four techniques for estimating a translation table $\mathbf{T_x}$ whose $(i,j)$-th element can be viewed as $p(w_i|f_i)$, the probability of a particular word $w_i$ given a visual-term $f_j$. In order to estimate the likelihood of each word for an unannotated visual-term occurrence vector $\vec{q}$, the word-likelihood vector $\vec{p} = \mathbf{T}_x\vec{q}$ can be calculated. The elements of $\mathbf{P}$, $p_i$, indicate the predicted likelihood of the word $w_i$.

Each of the four techniques assumed that the training matrices $\mathbf{W}$ and $\mathbf{F}$ (using the same definitions for $\mathbf{W}$ and $\mathbf{F}$ as in Section 2.2) have been weighted using inverse document frequency as described above.

The first technique, **Corr**, builds a correlation-based translation table such that $\mathbf{T}_{corr,0} = \mathbf{W}^{\mathrm{T}}\mathbf{F}$. The columns of $\mathbf{T}_{corr,0}$ are then normalised to sum up to unity, thus forming the table $\mathbf{T}_{corr}$.

The second technique, **Cos**, uses the overall occurrence pattern of each word and visual-term to estimate the similarity between each word and visual-term. These occurrence patterns are of course encoded in the columns of $\mathbf{W}$ and $\mathbf{F}$. The similarity of the column vectors can, amongst other techniques, be estimated through the cosine of the angle between the vectors. If we refer to the $i$-th column of the training matrices $\mathbf{W}$ and $\mathbf{F}$ by $\vec{w}_i$ and $\vec{f}_i$ respectively, we can calculate the cosine, $\cos_{i,j}$, as the angle between column vectors $\vec{w}_i$ and $\vec{f}_j$. If we now define $\mathbf{T}_{\cos,0}$ such that the $i,j$-th element is $\mathbf{T}_{\cos,0}(i,j) = \cos_{i,j}$, then we can calculate a translation table $\mathbf{T}_{\cos}$ by normalising the columns of $\mathbf{T}_{\cos,0}$ to sum to unity.

The final two techniques, **SvdCorr** and **SvdCos** are variations on the first technique; translation tables $\mathbf{T}_{corr,svd}$ and $\mathbf{T}_{cos,svd}$ are generated using the same approach as described above, but instead of using the raw training matrices $\mathbf{W}$ and $\mathbf{F}$, matrices $\mathbf{W}_{svd}$ and $\mathbf{F}_{svd}$ are used instead. $\mathbf{W}_{svd}$ and $\mathbf{F}_{svd}$ are formed by using the SVD to reduce the dimensionality (and thus noise) in the original matrices. In Pan *et al*'s work, the number of singular values retained for the reconstruction, $k$, is selected so that 90% of the variance of the distribution of singular values is preserved.[15]

### 2.4 Semantic Spaces

Our Linear-Algebraic Semantic Space approach[13] is a generalisation of a text-retrieval technique called Cross Language Latent Semantic Indexing,[19] which is itself an extension of Latent Semantic Indexing/Analysis (LSI/LSA).[18]

In general, any document (be it text, image, or even video) can be described by a series of observations, or measurements, made about its content. We refer to each of these observations as terms. Terms describing a document can be arranged in a vector of term occurrences, i.e. a vector whose $i$-th element contains a count of the number of times the $i$-th term occurs in the document. There is nothing stopping a term vector having terms from a number of different modalities. For example a term vector could contain term-occurrence information for both 'visual' terms and textual annotation terms. Given a corpus of documents, it is possible to form a matrix of observations or measurements (i.e. a term-document matrix), $\mathbf{O}$. Using the nomenclature adopted earlier, in our case $\mathbf{O} = [\mathbf{T}^{\mathrm{T}}|\mathbf{W}^{\mathrm{T}}]$.

Fundamentally, the Semantic Space technique works by estimating a rank-reduced factorisation of a term-document matrix of data, $\mathbf{O}$, into a term matrix $\mathbf{T}$ and a document matrix $\mathbf{D}$:

$$\mathbf{O} \approx \mathbf{TD} \ . \tag{6}$$

The two vector bases created in the decomposition form aligned vector-spaces of terms and documents. The rows of the term matrix, $\mathbf{T}$, create a basis representing a position in the space of each of the observed terms. The columns of the document matrix, $\mathbf{D}$, represent positions of the observed documents in the space. Similar documents and terms share similar locations in the space.

Assume that we have two collections of images; a training set with keyword annotations and a test set without. The content of each image can be represented by a vector of 'visual-term' occurrences. A cross-modality term-document matrix, $\mathbf{O}_{train}$ can be created for the training set of images by combining the visual-term occurrence vector with the keyword-term occurrence vector for each image. This can then be factorised according to Equation 6 into a term matrix $\mathbf{T}_{train}$ and a document matrix $\mathbf{D}_{train}$ by using the truncated singular value decomposition and letting $\mathbf{T} = \mathbf{U}_k$ and $\mathbf{D} = \mathbf{\Sigma}_k\mathbf{V}_k^{\mathrm{T}}$ (the $k$ refers to the number of singular values selected).

In order to make the unannotated test images search-able, we can project them into the semantic space described by $\mathbf{T}_{train}$ (and $\mathbf{D}_{train}$). Firstly, a cross-modality term-document matrix, $\mathbf{O}_{test}$ must be created for the test set of images by setting the number of occurrences of each (unknown) keyword to 0. It can be shown that it is possible to create a document matrix, $\mathbf{D}_{test}$ for the test documents as follows:

$$\mathbf{D}_{test} = \mathbf{T}_{train}^{\mathrm{T}} \mathbf{O}_{test} \ . \tag{7}$$

In order to query the test set for images relevant to a term, we just need to rank all of the images based on their position in the space with respect to the position of the query term in the space. The angle between the vectors or cosine similarity is a suitable measure for this task.

## 3. EXPERIMENTS

Our previous work[20] has demonstrated that the semantic space technique can be quite effective at image retrieval. In this paper we investigate and compare the performance of the semantic space technique, the linear transform technique, and the correlation approaches defined by Pan *et al.*[15] Each of the approaches is applied to the same image dataset using exactly the same features ($\mathbf{W}$ and $\mathbf{F}$ matrices). The techniques reliant on the use of a truncated SVD have their optimal dimensionality selected by maximising the overall mean average precision of a retrieval experiment on a validation set of images, as described below.

### 3.1 Dataset

The Corel dataset has been criticised in the past as both being "too easy", and as too small for proper retrieval evaluation.[21,22] However, that being said, it is still used as the *defacto* standard in most auto-annotation papers. In this study, we believe that the choice of this data-set is reasonable because the experiments will be repeatable and comparable. Also, we don't believe that the dataset is quite as easy as has sometimes been suggested since the state-of-the-art techniques struggle to annotate it effectively. One reason for this is that the dataset is actually quite representative of other real-world datasets in that it contains many errors, and strange keyword choices. These factors confound the problem of training a machine to learn how to annotate image content effectively, but are realistic of training data in the real world.

We have split the dataset into three subsets for experimental purposes; a 4000 image training set, a 500 image validation set, and a 500 image test set. The 500 image test set is the same as used in Ref. 7. As described above, the optimal number of dimensions for techniques using the SVD to reduce noise is selected such that the mean average precision, averaged over all possible queries, of a hypothetical retrieval experiment carried out on the validation set is maximised (with just the training set used for training). Once an optimal dimensionality has been found, the system is re-trained using both the training and validation data before being tested on the test data.

### 3.2 Image features

A large part of how well a particular technique is able to learn the relationships between and image and its semantics is directly a function of the descriptors or feature-vectors used to represent the pixel content of the image and build the visual-term representations. In this paper, we choose to use two different techniques for building visual-terms.

#### 3.2.1 Blobs

The blobs feature is the same as found in Ref. 7. The feature was created by segmenting each image into a number of blobs, and then calculating a descriptor using various colour, texture and shape attributes of each blob in the respective image. The set of descriptors was then clustered using k-means to create a vocabulary of 500 visual-terms. Each blob was then converted to a visual term by vector quantising its descriptor into the nearest visual term. Term-occurrence vectors were finally calculated by counting the occurrences of each visual term in each image.

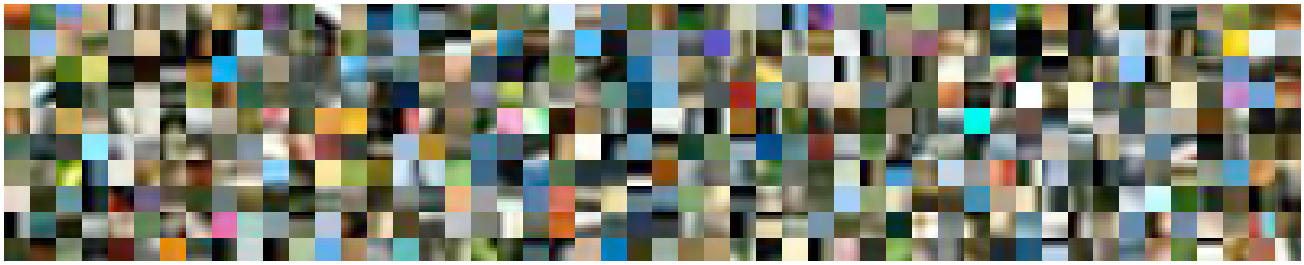


Figure 1. Example of a visual vocabulary from clustered DCT blocks

#### 3.2.2 Clustered DCT Features

Carneiro *et al*[12] demonstrated a system for automatic annotation that works well on the Corel dataset. One of the reasons suggested for this is that the Discrete Cosine Transform (DCT) features used in the work were particularly powerful. In this work we cannot use quite the same feature as[12] because of the requirement that we have discrete visual-terms, rather than a continuous feature; however, it is possible to use the DCT to create a visual vocabulary by clustering image blocks in the DCT domain, and then applying vector quantisation to create lists of visual-terms for each image. In our previous work we demonstrated such a feature, which we will again use here.[20]

In our implementation of a DCT-based feature, we split each image into a sequence of overlapping $8 \times 8$ blocks. We also left a 4-pixel border around the edge of each image in order to reduce the likelihood of problems occurring due to the black borders in many of the Corel images. For each of the Red, Green and Blue planes of the block we calculated the DCT, and ordered the DCT coefficients from highest to lowest frequency. It is well known that the lowest frequency coefficients are less important visually, so of the 64 DCT coefficients, we kept only the highest 10 coefficients from each plane (including the DC coefficient). The selected coefficients from each plane were appended together to form a feature vector for the respective image block.

Once sets of feature-vectors were calculated for each image, a random sample was drawn and clustered using K-means. The cluster centres formed a codebook, or vocabulary, of visual-terms which was then used to assign a visual term to each feature-vector by finding the closest term in the codebook (using Euclidean distance). An example of the representative image blocks found in a typical 500 term vocabulary generated from the Corel set is shown in Figure 1.

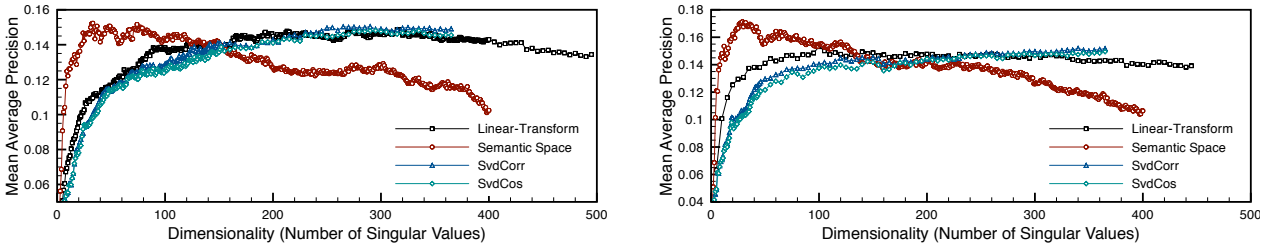
For the experiments described in this paper, we used the optimal feature settings found in Ref. 20: i.e. a vocabulary size of 500 visual-terms, and a maximum overlap of 6 pixels per block (the blocks were extracted using a sliding-window approach, moving the window by an offset of 2 pixels in each step).

### 3.3 Optimal dimensionality for reducing noise

In addition to selecting the number of singular values as those that preserve 90% of the variance for the Cos and Corr approaches,[15] we also tried to find an actual optimum value. The linear-transform and semantic space techniques also need to have a suitable number of dimensions chosen by optimising an objective function, such as the mean average precsion on the validition data. Figure 2 shows how the number of singular values selected during the truncated SVD of each approach affects the mean average precision of the validation data-set using the blob feature. The plots show that the SvdCos, SvdCorr and linear-transform techniques are all relatively insensitive to the choice of dimensionality, whereas the semantic space approach is relatively sensitive. This is important as any choice of dimensionality on the validation set will be sub-optimal when applied to the test-set; when the curve is flatter, this is less of an issue.

### 3.4 Semantic retrieval

Semantic retrieval performance of the different techniques and visual-terms features is compared by training each of the techniques using the training and validation sets and then using the trained keywords to attempt retrieval of the images in the test set. Since the ground truth annotations of the test set are known, it is possible to determine which images are relevant to a particular keyword query and thus calculate precision and recall.




(a) (b)

Figure 2. Plot of dimensionality ($k$) versus mean average precision of the validation data. (a) shows unweighted blob data; (b) shows the use of the idf weighting on the blob data.

Table 1 summarises the mean average precision for all the retrieval experiments. It should be noted that these precision-recall scores are perhaps a little misleading as each of the techniques perform unequally for different queries. The different machine learning approaches learn different underlying relationships between the visual-term features and the annotation keywords.

The results in Table 1 show a number of findings. Firstly, as expected from prior work the clustered DCT features perform better than the blob features for all annotation algorithms. The inverse document frequency weighting scheme applied to the blob features helps both the semantic space technique and linear-transform technique, but doesn't give much of an improvement for the correlation based methods. With the clustered DCT feature the weighting tends to hinder the performance consistently for all the techniques, albeit with differing amounts. The semantic space and linear-transform techniques are particularly badly affected by the weighting with the DCT feature. All the correlation techniques (Cos, Corr, SvdCos and SvdCorr) appear to perform very similarly to each other.

## 4. DISCUSSION

On the whole, there isn't much variation in performance amongst the techniques described, however, under certain conditions certain techniques perform with a considerable relative improvement. In particular both the linear-translation and semantic space approaches outperform the correlation-based approaches with the more complex DCT data. Conversely, with the unweighted blob data, the optimised SvdCos method works best.

The observation that the IDF weighting hampers the machine learning with the DCT data mirrors the findings in[20] where an entropy-based weighting scheme was found to boost semantic space retrieval with blobs, but hinders it when clustered DCT features are used. The results presented in this paper also back-up the assertion made in our previous work[20] that the semantic space technique performs better than a PLSA-based approach[23] even though the PLSA approach was shown to outperform an approach based on SvdCos and SvdCorr.[23]

Table 1. Summary of retrieval performance with the different methods.

| **Feature** | **Blob Feature** | | | | **Clustered DCT Feature** | | | |
|---|---|---|---|---|---|---|---|---|
| | **unweighted** | | **idf** | | **unweighted** | | **idf** | |
| | $k_{opt}$ | $mAP$ | $k_{opt}$ | $mAP$ | $k_{opt}$ | $mAP$ | $k_{opt}$ | $mAP$ |
| Linear-Transform | 315 | 0.144 | 100 | 0.164 | 275 | 0.189 | 425 | **0.184** |
| Semantic Space | 33 | 0.152 | 29 | **0.188** | 271 | **0.191** | 266 | 0.176 |
| Corr | n/a | 0.158 | n/a | 0.159 | n/a | 0.165 | n/a | 0.165 |
| Cos | n/a | 0.158 | n/a | 0.158 | n/a | 0.162 | n/a | 0.162 |
| SvdCorr | 90% of var | 0.158 | 90% of var | 0.159 | 90% of var | 0.167 | 90% of var | 0.165 |
| SvdCos | 90% of var | 0.157 | 90% of var | 0.157 | 90% of var | 0.164 | 90% of var | 0.163 |
| SvdCorr | 290 | 0.160 | 340 | **0.165** | 240 | 0.167 | 260 | 0.166 |
| SvdCos | 295 | **0.162** | 365 | 0.161 | 250 | 0.164 | 260 | 0.163 |

For a detailed understanding of how these techniques (in particular the semantic space approach) compare to other state-of-the-art techniques (e.g. Ref. 12, 24), the reader is encouraged to consult our previous work.[20] Whilst there are approaches that outperform the semantic space technique on the Corel dataset in terms of raw annotation performance, we have to bear in mind that techniques are only directly comparable if they use exactly the same image feature representation. In general, it appears that the semantic-space approach, and indeed the other techniques described in this paper which perform similarly, can currently be seen as the best methods to use when the image feature representation consists of discrete visual terms.[20] The techniques described in this paper have two particular advantages over many of the other state-of-the-art techniques; firstly they are relatively computationally inexpensive (for example, Carneiro *et al*[12] reported that annotating the Corel set on a 3000 node cluster could take about 1 hour; extracting DCT visual-terms and building a semantic space on the other hand takes less than 20 minutes on a single workstation). Secondly, the methods presented in this paper are deterministic, unlike may of the other state-of-the-art algorithms which often have random components, or rely on the algorithms, such as EM, which can be prone to getting stuck in local minima in the high dimensional spaces associated with image features.

### 4.1 Computational cost

The **Cos** and **Corr** techniques of Pan *et al*[15] obviously have the least amount of computational complexity of all the methods as they do not require computation of an expensive factorisation. However, the gain in computational performance is offset by the relatively worse retrieval and annotation abilities. The standard Lanczos techniques for calculating the SVD have a time complexity of $O(pqr^2)$, where $p$, $q$ and $r$ are the number of rows, columns and desired singular triples (singular values and corresponding left and right singular vectors) respectively.[25] The SvdCos and SvdCorr techniques require two decompositions each, and experimental results show they tend to require relatively large dimensionality, $r$, compared to the small number of dimensions required by the semantic space technique in some cases. The linear transform sits in-between these techniques in terms of time complexity (relatively large number of dimensions for some features, but only one SVD is required). The SVD carried out for the semantic space approach is applied to a larger matrix than for the other techniques, however because fewer singular values are required, the decomposition actually requires less time than the other techniques. Modern incremental SVD techniques can often have a lower time complexity, however, the number of dimensions is still an important factor.[25]

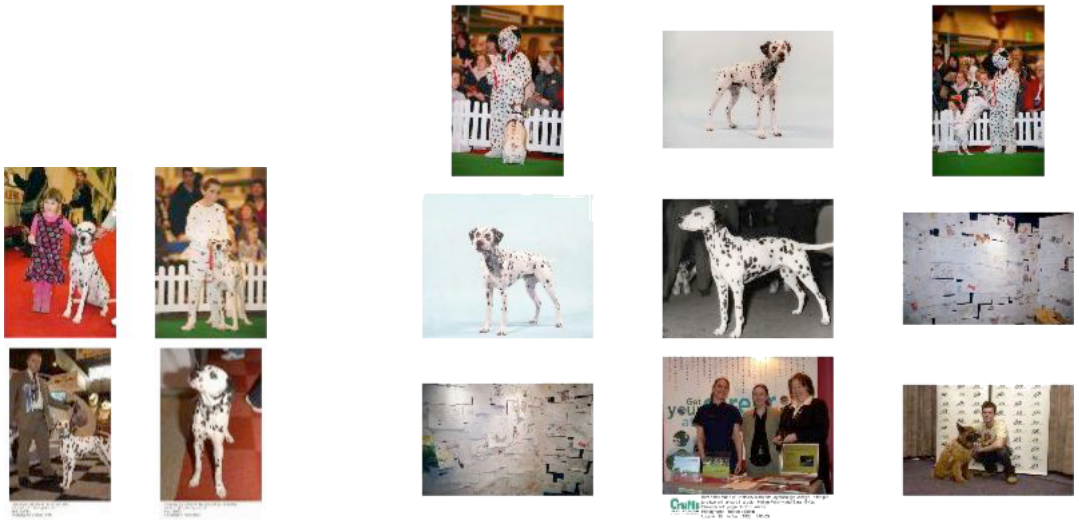
### 4.2 Practical real-world semantic retrieval

Very few papers on automatic annotation techniques discuss the applicability of the proposed technique to real-world problems and data-sets, and instead just concentrate on the performance against standard test sets. Whilst this is an important part of the scientific process, it can often be instructive to experiment with data-sets where there is a real problem to be solved.

In the course of our research, we have worked closely with many organisations and archives that deal with image search on a daily basis. By collaborating with the picture librarians at these organisations we have collected sample queries, meta-data and imagery, which has enabled us to analyse some of the problems associated with image retrieval. This analysis has led to a much greater understanding of the problem of the semantic gap from the point-of-view of the image librarians and also from the point-of-view of the researchers in computational image retrieval.[4]

In order to investigate the power of our semantic space technique in the real world, we applied the technique to a test collection of images obtained from the Kennel Club picture library. In total, we have a collection of 7120 images, but only 2703 of the images are annotated with subject meta-data in the form of keywords. In order to represent the visual features of the images, visual terms were created by quantizing SIFT features from salient regions detected by the difference-of-Gaussian approach.[9,14] The visual vocabulary was set to a size of 3000 terms, and there was a total of 2003 distinct annotation keywords/phrases. Training of the semantic space was performed using all of the annotated images, and all of the unannotated images were folded into the resultant space.

Performance of the search technique on the Kennel Club data-set is difficult to judge quantitatively due to the lack of ground-truth annotations from which to compare performance. However, we can make a number of



(a) (b)

Figure 3. Searching for "dalmatians". (a) Annotated training images; (b) Top 9 ranked retrieved (unannotated) images (rank increases left-to-right, top-to-bottom). Images Copyright © 2009, The Kennel Club Picture Library, All rights reserved.

observations. The first observation is that the retrieval performance can vary dramatically depending on the query — some queries appear to work very well, with many relevant images retrieved near the top of the ranked list, however other queries seem to return almost random results. The reasons for this are two-fold;[26] firstly, the training set may be deficient, and not contain enough exemplars of a given concept to accurately learn it's visual attributes (if any), and secondly, the choice of visual feature can have a large impact on performance. For example, we would not expect our gray-level SIFT-based visual terms to be able to learn relationships between colours. The discriminability of the visual features is also important as we may find a number of unrelated textual terms represented by a set of similar visual features.

The 'Dalmatian' query illustrated in Figure 3 is an example of a query that works reasonably well. We hypothesize that one of the reasons for this is that the visual features represent 'spots' quite well. All of the top-ranked images include multiple dark spot-like features on a lighter background, just like the spots on a Dalmatian, which indicates that the space has encoded a strong correlation between these spot-like features and the word 'Dalmatian'. Another interesting fact is that there are actually more than four training images depicting Dalmatians in the training set, however, their key-wording has been misspelled as 'Dalmation'. If we search the semantic space using the misspelled term 'Dalmation' as a query, we interestingly get back the same images as if we had spelled it correctly (albeit in a slightly different order). This indicates that the terms 'Dalmatian' and 'Dalmation' both occur at very similar points in the semantic space even though they do not co-occur in any of the training data.

Another query of interest is that for the term 'agility'. Generally speaking 'agility' does not have any particular visual meaning, however, in the context of the Kennel Club data-set, it tends to refer to images of a particular event at a dog show where dogs (and their owners!) run and jump over and though obstacles. The semantic space performs quite well at retrieving these images. We hypothesise that it is a non-trivial combination of visual features that enables these images to be associated with agility. For example, strong line-like features from the obstacles, and fur texture on the dogs.

The above discussion is rather qualitative due to the lack of ground-truth. We were however able to get a more quantitative idea of the performance by trying a number of queries and seeing how many results looked relevant. In total we tried 31 queries, and analysed the first 20 results returned. The queries used were taken from a set of actual request recorded by the Kennel Club's picture librarian. If we take into account all of the first 20 result images, then the overall precision $P_{20}$ (calculated as the number of relevant images in the top 20 divided by 20) is around 8.7%. This number seems low, but we have to bear in mind that we do not actually

know whether there were any relevant images in the set (or indeed how many relevant images there may have been), so it is quite possible that in a number of cases the fact that no relevant images were retrieved is purely down to the fact that there were no relevant images for the query. Of the 31 queries, only 12 actually returned any relevant results (the number of relevant results varied between 1 and 13).

## 5. CONCLUSIONS AND FUTURE WORK

This paper has presented a new technique for automatic annotation and semantic retrieval. The new linear-transform technique can perform well when compared to a similar class of techniques. The semantic space proposed in previous work performs quite similarly to the linear-transform technique, however the linear-transform approach has a small advantage in that the space into which unannotated images are projected has actual semantic keyword terms as its axes. Contrast this to the semantic space approach where the axes of the space are linear combinations of both textual and visual terms. Having the keyword terms as axes gives two possible advantages; firstly, the space is easier to understand and search. Secondly keyword independence is enforced, which can be useful if the keywords are known not to be synonyms.

Our plans for future work currently revolve around two important areas; firstly we wish to develop better visual representations. This will involve both the fusion of more, different, types of visual features, but also the development of better visual-term representations. In particular, current visual-term representations are more synonymous with letters rather than words when compared to a human language — that is to say that the individual visual-terms don't really have any semantic meaning. However, groups of visual-terms in a particular spatial arrangement may well have a semantic meaning. If these groups of visual-terms can be extracted automatically, then a better, more semantically justified visual-term representation may be developed.

Secondly, we wish to explore how semantic retrieval techniques can be scaled-up to collections containing perhaps hundreds-of-thousands, or even millions of images. In particular, we wish to explore how iterative learning techniques can be applied to building semantic retrieval systems, rather than trying to train the system in a batch mode.

## 6. ACKNOWLEDGMENTS

This work was supported by the European Union under the Seventh Framework project LivingKnowledge (IST-FP7-231126), and the LiveMemories project, graciously funded by the Autonomous Province of Trento (Italy).

## REFERENCES

[1] Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., and Jain, R., "Content-based image retrieval at the end of the early years," *PAMI* **22**(12), 1349–1380 (2000).

[2] Enser, P. G. B., Sandom, C. J., Hare, J. S., and Lewis, P. H., "Facing the reality of semantic image retrieval," *J. Doc.* **63**, 465–481 (August 2007).

[3] Enser, P., "The evolution of visual information retrieval," *J. Inf. Sci.* **34**, 531–546 (August 2008).

[4] Hare, J. S., Lewis, P. H., Enser, P. G. B., and Sandom, C. J., "Mind the gap," in [*Multimedia Content Analysis, Management, and Retrieval 2006*], Chang, E. Y., Hanjalic, A., and Sebe, N., eds., **6073**, 607309–1–607309–12, SPIE, San Jose, California, USA (January 2006).

[5] Mori, Y., Takahashi, H., and Oka, R., "Image-to-word transformation based on dividing and vector quantizing images with words," in [*MISRM '99*], (1999).

[6] Jeon, J., Lavrenko, V., and Manmatha, R., "Automatic image annotation and retrieval using cross-media relevance models," in [*SIGIR '03*], 119–126, ACM Press, New York, NY, USA (2003).

[7] Duygulu, P., Barnard, K., de Freitas, J. F. G., and Forsyth, D. A., "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary," in [*ECCV '02*], 97–112, Springer-Verlag, London, UK (2002).

[8] Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D. M., and Jordan, M. I., "Matching words and pictures," *J. Mach. Learn. Res.* **3**, 1107–1135 (2003).

[9] Sivic, J. and Zisserman, A., "Video google: A text retrieval approach to object matching in videos," in [*ICCV*], 1470–1477 (October 2003).

[10] Hare, J. S. and Lewis, P. H., "On image retrieval using salient regions with vector-spaces and latent semantics.," in [*CIVR*], Leow, W. K., Lew, M. S., Chua, T.-S., Ma, W.-Y., Chaisorn, L., and Bakker, E. M., eds., *LNCS* **3568**, 540–549, Springer (2005).

[11] Datta, R., Joshi, D., Li, J., and Wang, J. Z., "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv.* **40**(2), 1–60 (2008).

[12] Carneiro, G., Chan, A. B., Moreno, P. J., and Vasconcelos, N., "Supervised learning of semantic classes for image annotation and retrieval," *PAMI* **29**(3), 394–410 (2007).

[13] Hare, J. S., Lewis, P. H., Enser, P. G. B., and Sandom, C. J., "A Linear-Algebraic Technique with an Application in Semantic Image Retrieval," in [*CIVR 2006*], Sundaram, H., Naphade, M., Smith, J. R., and Rui, Y., eds., *LNCS* **4071**, 31–40, Springer (2006).

[14] Lowe, D., "Distinctive image features from scale-invariant keypoints," *IJCV* **60**, 91–110 (January 2004).

[15] Pan, J.-Y., Yang, H.-J., Duygulu, P., and Faloutsos, C., "Automatic image captioning," *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on* **3**, 1987–1990 Vol.3 (27-30 June 2004).

[16] Eckart, C. and Young, G., "The approximation of one matrix by another of lower rank," *Psychometrika* **1**, 211–218 (1936).

[17] Monay, F. and Gatica-Perez, D., "On image auto-annotation with latent space models," in [*ACM MM '03*], 275–278, ACM Press (2003).

[18] Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A., "Indexing by latent semantic analysis," *Journal of the American Society of Information Science* **41**(6), 391–407 (1990).

[19] Landauer, T. K. and Littman, M. L., "Fully automatic cross-language document retrieval using latent semantic indexing," in [*Proceedings of the Sixth Annual Conference of the UW Centre for the New Oxford English Dictionary and Text Research*], 31–38 (October 1990).

[20] Hare, J. S., Samangooei, S., Lewis, P. H., and Nixon, M. S., "Semantic spaces revisited: investigating the performance of auto-annotation and semantic retrieval using semantic spaces," in [*ACM CIVR '08*], 359–368, ACM (July 2008).

[21] Tang, J. and Lewis, P. H., "A study of quality issues for image auto-annotation with the corel dataset," *IEEE Trans. Circuits Syst. Video Techn.* **17**(3), 384–389 (2007).

[22] Müller, H., Marchand-Maillet, S., and Pun, T., "The truth about corel - evaluation in image retrieval.," in [*CIVR*], Lew, M. S., Sebe, N., and Eakins, J. P., eds., *LNCS* **2383**, 38–49, Springer (2002).

[23] Monay, F. and Gatica-Perez, D., "Modeling semantic aspects for cross-media image indexing," *PAMI* **29**(10), 1802–1817 (2007).

[24] Yavlinsky, A., Schofield, E., and Rüger, S., "Automated Image Annotation Using Global Features and Robust Nonparametric Density Estimation," in [*CIVR 2005*], Leow, W. K., Lew, M. S., Chua, T.-S., Ma, W.-Y., Chaisorn, L., and Bakker, E. M., eds., *LNCS* **3568**, 507–517, Springer, Singapore (2005).

[25] Brand, M., "Incremental singular value decomposition of uncertain data with missing values," in [*In ECCV*], 707–720 (2002).

[26] Hare, J. S., Lewis, P. H., Enser, P. G. B., and Sandom, C. J., "Semantic facets: an in-depth analysis of a semantic image retrieval system," in [*ACM CIVR '07*], 250–257, ACM Press, New York, NY, USA (2007).