entropy

MDPI

Article

An Informed Framework for Training Classifiers from Social Media [†]

Dong Seon Cheng [1,*] and Sami Abduljalil Abdulhak [2]

[1] Department of Computer Science and Engineering, Hankuk University of Foreign Studies, 81 Oedae-ro, Mohyeon-myeon, Cheoin-gu, Yongin-si, Gyeonggi-do 449-791, South Korea

[2] Department of Computer Science, University of Verona, Strada Le Grazie 15, I-37134 Verona, Italy; sami.naji@univr.it

[*] Correspondence: cheng_ds@hufs.ac.kr; Tel. +82-31-330-4366

[†] This paper is an extended version of our paper published in 18th International Conference on Image Analysis and Processing (ICIAP 2015), Genoa, Italy, 7–11 September 2015.

Academic Editor: Andreas Holzinger
Received: 18 January 2016; Accepted: 28 March 2016; Published: 9 April 2016

Abstract: Extracting information from social media has become a major focus of companies and researchers in recent years. Aside from the study of the social aspects, it has also been found feasible to exploit the collaborative strength of crowds to help solve classical machine learning problems like object recognition. In this work, we focus on the generally underappreciated problem of building effective datasets for training classifiers by automatically assembling data from social media. We detail some of the challenges of this approach and outline a framework that uses expanded search queries to retrieve more qualified data. In particular, we concentrate on collaboratively tagged media on the social platform Flickr, and on the problem of image classification to evaluate our approach. Finally, we describe a novel entropy-based method to incorporate an information-theoretic principle to guide our framework. Experimental validation against well-known public datasets shows the viability of this approach and marks an improvement over the state of the art in terms of simplicity and performance.

Keywords: training sets; image classification; Shannon entropy; social media

1. Introduction

The last decade has seen an explosive growth of on-line media-sharing communities, with a consequential drastic increase in multimedia resources freely available on the web for companies and researchers to study in detail. Far bigger in volume than anyone could collect or assemble, this mass of data voluntarily supplied by millions of people is also difficult to organize and understand. Innovative methods have recently been used to automate these tasks, so much so that news coverage is talking about artificial intelligence finally becoming mainstream.

Aside from machine learning tools becoming useful to the public at large in a very visible way, their success also lies in how social media has helped by providing a critical mass of data for their training and validation. In particular, there has been a remarkable step forward in image classification and object recognition (especially face recognition)—classical machine learning problems. There is a wealth of complex information buried in the data, with clues provided by the suppliers, the social media users, but extracting this information is difficult because such clues are often inaccurate, wrong, or ambiguous. In other words, the data is largely unstructured and loosely labeled (see Figure 1).

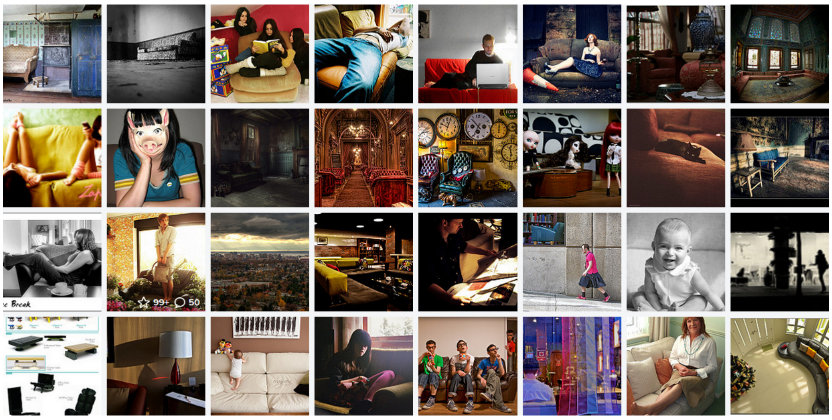


Figure 1. Montage of a few images returned by a simple search for "sofa" on Flickr. Most of these are not representative of the sofa class, either because they are marginally or loosely related (e.g., people sitting on sofas, or sofas present in the background).

One way to deal with these challenges is to select more narrow and qualified data from a larger pool. In this work, we focus on the problem of building effective training sets for object recognition, with the larger goal of providing an automated framework for generating *ad-hoc* classifiers. This is a generally important problem, although less appealing than themes like feature design and model learning. Mostly studied in the past [1], crowd-funding platforms like the Amazon Mechanical Turk [2] are symbolic of the way forward. However, these approaches still require a considerable effort by the researchers, both in temporal and monetary terms, and are generally restricted to a few categories. A notable exception is the ImageNet project [3], which mirrors the taxonomy of Wordnet [4], with more than 100,000 synsets (meaningful concepts represented as synonym sets). Unfortunately, many of the less-used synsets contain very few images, thus making their use as training sets generally unfeasible.

Our approach towards building training sets focuses on the study of textual tags (short labels) provided by the users when sharing their images (for example, on Flickr [5]). These cues to the image content are recognized to be often unreliable, since they are freely entered and not associated to any ontology or structured categorization. Moreover, there are often complex motivations involved in the labeling [6,7], and therefore tags do not necessarily always describe the content of the images [8]: on Flickr, which is often used by photography lovers, many tags note the camera device (by brand and model) and the photo shoot setup (lighting and effects). In the framework we propose, our strategy is based on the mechanism of *query expansion* [9], which first builds statistics of the tags associated with a given target class, then filters this list to remove potentially noisy tags, and finally uses the remaining tags to qualify multiple search queries that assemble the eventual training set.

Compared to other approaches in the literature, our framework is simple but effective, giving excellent results that are validated on different image classification datasets, like Pascal Visual Object Classes (VOC) [10,11] and ImageNet [12]. Although there are some drawbacks, which we discuss in the experimental section, it highlights very clearly how useful the wisdom of the crowds is: with a set of images collected automatically in 15–20 min, we obtained more than 81% of average precision on the Pascal VOC 2012 classification problem. We also improve on OPTIMOL [13], which exploits the visual content of the images. Finally, we introduce a novel tags selection method based on the Shannon entropy that incorporates information-theoretic principles in our framework, and show its viability experimentally. In future works, we envision an improved framework informed not only by metadata information, but also visual and textual cues, in order to develop a theory of visual knowledge.

The rest of the paper is organized as follows: Section 2 presents some related literature and remainders for the query expansion strategy; Section 3 describes our framework, while Section 4 discusses validation experiments. Finally, conclusions and directions for future work are presented in Section 5.

2. State of the Art

Metadata-rich social media have been used quite often in recent years to improve on traditional recognition tasks: for example, in [14], Flickr groups are used to learn accurate image similarities. They have become a viable alternative to harvesting data from the web at large [15–17]. Like the comparable works dealing with the automatic generation of training sets [13,15,18,19], our approach focuses on the task of capturing a range of diverse but consistent picture representatives for a given visual concept (like dog, car, *etc*.). In other words, our idea is to design a system that creates *visual synsets* [3], like in ImageNet, but with minimal intervention, and fast, dynamic, and customizable structure (for concepts outside WordNet [20]).

In [13,18], the approach exploits seed images, manually selected or retrieved by an image search engine, as a set of trusted representatives for an object class, and then iteratively enlarges this set by learning a class model and using it to evaluate and add new images to the set. It aims at mimicking how humans build knowledge incrementally, but it also shares one weakness, namely that noisy and misleading seed images may derail the process, causing it to reinforce an erroneous characterization.

Thus, one challenge is reducing possible false positive images. Since widely-used search engines like Google are queried using textual keywords, it is possible to indirectly control the image retrieval by using qualified keywords [15] or bi-grams containing the entity of interest [19]. In this last work, a search keyword (plus an hyperonymy specifying the context) is used to generate a set of bi-grams, each one of them addressing a specific semantic aspect of the visual concept. The bi-grams are produced by looking at Google Ngrams (See https://books.google.com/ngrams/info), individuating those additional terms that are visual adjectives (where the "visual" characteristics are found by WordNet), present participles (found by basic natural language processing methods), and hyponymy. These bi-grams, ordered by their original frequency in the Ngrams repository or uniformly weighted, are used to create specific sets of images (or classifiers) that once pooled together give the final dataset (or classifier). Selecting visually-representative keywords is an important issue that is not easy to solve: in [21], the visually-representative tags are identified by their high use for images similar in content. An alternative to using human intelligence to define visual knowledge is to mine the web [16,17] for this information.

All of these automatic approaches rely on indexing methods, like the Google image search, which are not open source, meaning that one step of the collection consists in a black box, where the cues used to gather images are hidden and changing over time (due to advancements in the search engine and addition of new indexed content). As a consequence, the performance of such approaches may vary considerably, with no guarantee of repeatability.

3. Proposed Framework

Our framework exploits the fact that images on a media-sharing platform like Flickr can be labeled with multiple keywords, called *tags*. These are free-form text strings, but are generally short and mostly made of one word. Their intended use is to help navigating and searching collections of photographs, but there are often meaningless tags, shortcut tags, tags in languages different than English, *etc*. In other words, tags are noisy labels that might be misleading regarding the image content [8].

When performing image searches on Flickr, there is a variety of parameters that can be set. In our framework, we use the *search by tags* mode, where we can specify one or more tags to be required in the retrieved images (and, additionally, one or more tags we require not to be present). The default behavior of the Flickr search engine is to return the images sorted by relevance, where it is unclear how exactly the sorting is achieved. In other words, it acts as a less than fully reliable oracle.

The goal of our approach is to generate a training dataset of images for a given target object, of which we only know the name. We consider a dataset generated by assembling the images retrieved using the object name to be the *baseline* against which to compare. It is simple to do, performs reasonably well on some objects due to the Flickr oracle, but is prone to include misleading images and fails to take into account the noisy nature of tags. We contrast this to the informed choices in our framework.

3.1. Class Dictionaries

We call a *class dictionary* the collection of all tags associated with a given target object. Specifically, given a target object name x, a simple search on Flickr returns a set of images $\text{flickr}(x) = \{\mathbf{I}_k\}$, each accompanied by a set $W_k = \text{tags}(\mathbf{I}_k)$ containing the associated tags. The collection of all tags $T = \bigcup_k W_k$ then represents a crowd-based semantic characterization of the target class. For each word $w_i \in T$, we also collect its frequency of use f_i. Since the raw list often contains undesirable strings, we use a sequence of preprocessing steps to prune it: converting all tags to lower case, splitting space-delimited multi-words, removing numerical tags, non-English words and stopwords.

After obtaining the set T_{clean} of pruned tags, we consider two different types of dictionaries: the *full* class dictionary D_F and the *keyword position* class dictionary D_{KP}. The former simply contains all tags ($D_F = T_{clean}$), while the latter includes only the tags that appear before the object keyword in a given image tags list; that is, W_k is cut short when the class keyword is found. This choice is inspired by the fact that users tend to annotate dominant objects first [22], so we assume that tags earlier in a list are more likely to be about the prominent objects in an image.

3.2. Dictionary Filters

The next step in our framework is the selection of a limited number of relevant tags from the class dictionaries in order to perform query expansion. We consider the following different strategies to determine a list of N tags $E = \{w_i\}_{i=1}^{N}$.

- Frequency filter: this strategy simply sorts the full class dictionary in descending order of frequency f_i and then takes the first N tags, presumably indicative of widely shared visual semantics. Thus, concepts that are unrelated or occasionally related by context are ignored because they are lower in the ranking. For example, "Marie" might be used as a tag for an image of a dog, representing its owner, but its frequency in a large collection of images most likely will be low, as it is very situational. On the other hand, "Saint Bernard" might be higher in rank, as it is the name of a dog breed. The importance of tags by their frequencies is used in many applications [23–25].
- Keyword Position filter: similar to the previous filter, but starting from the keyword position class dictionary. The tags are sorted and the first N in the ranking are kept.
- Quality filter: here we exploit a semantic oracle developed in [26], which is essentially a list of 150 English terms considered by linguistic researchers to be "semantically rich and general", covering a wide variety of descriptions for different entities. The quality filter takes the intersection between the full class dictionary and the oracle list, and keeps the first N tags when sorting by frequency.
- Noun filter: this filter intersects the full class dictionary with a set of nouns that are in the hyponym sets of the given keyword, or in its immediate hyperonym set (found by the help of WordNet). Again, the remaining tags are sorted and the first N in the ranking are kept.

3.2.1. Entropy-Based Selection

All the dictionary filters described above follow the same principle for ordering and selecting the best tags: higher frequency of use. However, this principle does not take into account the interplay between the different tags. Simply put, they are not independently associated with pictures, but often their presence is correlated and inter-dependent.

If we model the presence or absence of a tag t as a Bernoulli random variable y_t, where $y_t = 1$ if the tag is associated with a given image, then a simple selection based on tag frequencies corresponds to sorting the tags in descending order of probability $P(y_t = 1)$, or, more precisely $P(y_t = 1|x)$, the probability conditioned to the search being about a specific class x. This particular conditioning can be dropped from the equations if we deal, as we do, with each class independently.

In fact, each image with multiple tags can be seen as a probabilistic event, representing a sample of the joint probability distribution $p(y_{t_1}, y_{t_2}, \ldots)$ of all the possible tags. Considering only the marginal probabilities $p(y_t)$ neglects to take into account the dependency between tags. For example, in an extreme case, two high probability tags that are always used together would both be chosen by frequency selection, but only one would bring in valuable information, and the other would be superfluous. Since it is not feasible to consider all possible tags, we can limit ourselves to the m tags most associated with a certain class, and thus study the joint probability mass function $p(y_{t_1}, y_{t_2}, \ldots, y_{t_m})$.

The most interesting issue at this point is which of the m tags brings in the most *information*? The information-theoretic answer is the random variable with the highest *entropy* (see [27] for related issues). For the Bernoulli variable y_t, the Shannon entropy is given by

$$H(y_t) = -p_t \log_2 p_t - (1 - p_t) \log_2 (1 - p_t) \tag{1}$$

where $p_t = P(y_t = 1)$ and the entropy are measured in bits. This entropy is a concave function of p_t, with a maximum for $p_t = 0.5$ and zero when $p_t = 0$ or $p_t = 1$. Thus, tags appearing half of the time contain more information than frequent or rare tags. For example, if a tag is always associated with a certain class, then it is generally useless for expanded searches (it will return roughly the same results).

Once we select the first tag t_1 with maximal entropy, the next most informative one is given by *relative entropy*:

$$\begin{aligned} H(y_t|y_{t_1}) &= -\sum_{(y_t, y_{t_1})} p(y_t, y_{t_1}) \log_2 p(y_t|y_{t_1}) && (2) \\ y_{t_2} &= \arg\max_{y_t} H(y_t|y_{t_1}) && (3) \end{aligned}$$

where $p(y_t, y_{t_1})$ is the joint probability mass function of selectable variable y_t and the already chosen variable y_{t_1}, and $p(y_t|y_{t_1})$ is their conditional probability mass function.

In general, given a set $Z_k = \{y_{t_1}, y_{t_2}, \ldots, y_{t_k}\}$ of k selected tags, we can select the next tag with

$$\begin{aligned} H(y_t|Z_k) &= -\sum_{(y_t, Z_k)} p(y_t, Z_k) \log_2 p(y_t|Z_k) && (4) \\ y_{t_{k+1}} &= \arg\max_{y_t} H(y_t|Z_k) && (5) \end{aligned}$$

After every selection, the relative entropy of the remaining variables will be reduced, so that we can stop the selection process either because we reach a given number of selected tags, or the relative entropy gets lower than a certain threshold. Both of these stopping criteria solve the only real issue with these calculations, which is that the state space (y_t, Z_k) grows exponentially, as 2^{k+1}. Experimentally, we rarely went beyond $k = 15$, where the state space has 32,768 possible combinations for the presence/absence of tags (it takes a few minutes to calculate on a desktop).

An important property of the relative entropy is the chain rule:

$$\begin{aligned} H(y_{t_2}, y_{t_1}) &= H(y_{t_2}|y_{t_1}) + H(y_{t_1}) \\ H(y_{t_3}, y_{t_2}, y_{t_1}) &= H(y_{t_3}|y_{t_2}, y_{t_1}) + H(y_{t_2}, y_{t_1}) \\ \ldots &= \ldots \\ H(y_{t_{k+1}}, Z_k) &= H(y_{t_{k+1}}|Z_k) + H(Z_k). \end{aligned} \tag{6}$$

By this rule, we can calculate the total entropy of the final selection of tags as the sum of all the relative entropies calculated at each step and the initial marginal entropy of the first choice. In the end we have a list of N tags ranked by how informative they are, together with a measure $h_i = H(y_{t_i}|Z_{i-1})/H(Z_N)$ of their informativeness, obtained by normalizing each tag's entropy by the total entropy.

3.3. Query Expansion

With a list of N tags selected according to their relevance, the final step in our framework is the generation of expanded image search queries for downloading the training set pictures. Each expanded query is a straightforward request to the social media platform to retrieve pictures that are tagged both with the original class keyword x and the given tag t. For example, in Flickr, an advanced search with a comma delimited sequence "x, t" in *all* tag mode (meaning all terms must be used as tags) returns only images tagged with both x and t. Depending on the search engine capabilities and other contextual necessities (we discuss a few in the experiments), it is possible to use more complex queries to discard some images at the source.

Thus, for each of the N tags t_i, let $V_i = \text{flickr}(x, t_i)$ be the images returned by an expanded query. Since each of these requests to the social media platform are done independently, it is likely that some pictures will be returned multiple times, depending on how large the image pool is and how the search engine ranks the returns. An easy workaround is to discard the duplicates, as they will introduce biases for the classifiers. The entropy-based selection method tends to reduce the number of duplicates, as it takes into account the inter-dependency between tags.

Finally, the training set for class x is made by assembling all the images $\bigcup_{i=1}^{N} V_i$ of the expanded queries. Details about the total cardinality of the training set and the cardinality of each set V_i can be chosen depending on the experimental setup.

4. Experiments

In the previous section, we have outlined how our proposed framework can assemble training sets of pictures for image classification from social media platforms like Flickr. A qualitative assessment only goes so far in validating the effectiveness of the results (see the end of the article and the Supplementary Material). For a quantitative evaluation, in the first set of experiments we follow the methodology of [19], called simply *Semantic Trainer*, for testing classification performances and generalization capabilities. A second set of experiments follows the methodology of OPTIMOL [18]. Finally, the last set of experiments shows the viability of the novel entropy-based selection method.

4.1. Comparison against the Semantic Trainer

The Semantic Trainer approach tests image datasets by the conventional route of extracting features and training classifiers. These are then tested against the Pascal VOC 2012 image classification task [10]. In particular, using the MatConvNet toolbox (v1.0-beta17) [28], a convolutional neural network (CNN) [29] pre-trained on ImageNet is used to extract a 4096-dimensional feature vector for each image (from the FC7 layer of the network). Then, linear support vector machines (SVM) are trained on these features to learn class models, using the features from the class images as positives and the remaining features from the other classes as negatives. Finally, the classifiers are tested on the features extracted from the Pascal VOC 2012 validation datasets, and the interpolated average precision (AP) values are calculated (see [19] for further details).

Since Pascal VOC 2012 contains 20 class datasets, two of which the Semantic Trainer was unable to process, we also focus on the 18 remaining classes: *aeroplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, dog, horse, motorbike, person, sheep, sofa, train, and tv monitor*. In the first set of experiments, we compare the performance of our strategies (except the entropy-based selection) against the strategies of the Semantic Trainer (see Table 1). For each of the 18 classes, following our framework, we first create class dictionaries from 500 tagged images retrieved by a simple search, and then select $N = 10$ tags according to our dictionary filters. Using query expansion, we assemble a dataset of images with the same cardinality as the Pascal VOC 2012 ones (to keep it fair), and uniformly split between queries. That is, each query contributes to $1/10$ of the images in the final dataset.

In Table 1, we report the performances of the frequency filter (*freq-f*), the quality filter (*qual-f*), the keyword filter (*keyw-f*), and the noun filter (*noun-f*) against the simple search (*Flickr*) and the

Semantic Trainer strategies. The Semantic Trainer originally retrieved images using bi-gram (class keyword plus additional term) queries on the Google image search engine, and it has been modified here to work with Flickr. Thus, for each class, ten bi-gram queries are selected according to different strategies: *basic*, hyponym-based (*hypo*), verb-based (*verb*), visual adjective-based (*vadj*), combination of modules by bi-gram frequency (*fcom*), and combination of classifiers (*ccom*).

Table 1. Comparative classification results on the PASCAL VOC 2012 dataset between a simple search (Flickr column), the Semantic Trainer strategies, and our framework (see text for details). The numbers indicate interpolated average precision (AP), with the average AP in the last row (Bold font indicates the best for each class).

Classes	**Flickr**	**Semantic Trainer**						**Our Framework**			
		basic	***hypo***	***verb***	***vadj***	***fcom***	***ccom***	***freq-f***	***qual-f***	***keyw-f***	***noun-f***
aeroplane	97.7	97.3	95.2	97.4	97.1	**97.9**	97.3	97.3	96.4	97.3	93.6
bicycle	**82.8**	82.5	70.4	79.3	83.6	82.2	81.2	83.1	76.6	82.3	76.5
bird	90.7	90.4	91.5	89.9	90.2	90.1	91.7	90.7	**92.7**	89.3	92.0
boat	88.7	88.2	88.8	87.8	86.9	89.5	89.2	88.9	87.5	**89.3**	89.0
bottle	57.3	56.7	57.5	55.7	55.8	57.6	**58.3**	57.3	54.9	56.8	55.6
bus	**93.8**	93.7	87.3	94.3	93.0	94.1	93.0	93.4	91.6	93.1	92.8
car	72.6	75.6	69.8	71.9	**75.9**	71.6	74.7	73.2	74.6	73.9	73.2
cat	91.5	89.1	92.9	90.6	90.9	91.4	**93.1**	92.9	89.6	90.0	91.5
chair	70.3	69.9	73.3	71.1	72.3	67.8	**74.3**	68.9	66.5	71.0	59.5
cow	**79.0**	73.9	73.6	71.8	75.1	75.7	77.7	76.6	78.8	76.1	64.9
dog	88.9	87.3	89.5	87.3	87.1	86.1	**89.7**	88.8	88.7	86.6	87.5
horse	**85.1**	76.8	80.1	76.9	81.7	80.5	83.0	84.8	83.8	82.2	80.7
motorbike	89.1	89.4	4.7	88.9	91.0	90.7	**91.3**	89.1	89.8	85.5	79.2
person	60.4	61.5	**72.8**	60.6	58.1	63.9	68.4	57.8	71.8	66.8	58.1
sheep	84.9	84.0	85.6	82.9	85.2	84.9	87.2	84.9	**86.3**	86.2	79.5
sofa	58.0	59.6	45.7	52.7	58.7	58.2	59.0	10.6	**62.7**	49.8	39.1
train	92.8	92.4	90.6	93.1	92.2	**93.6**	93.2	89.1	92.7	91.8	91.0
tv monitor	25.0	74.1	55.4	26.2	45.0	46.8	53.1	73.4	**77.1**	31.5	69.3
Mean AP	78.3	80.15	73.6	76.6	78.9	79.1	80.9	77.8	**81.3**	77.8	76.3

These results indicate comparable performances between our framework and the Semantic Trainer strategies, with the difference that our approach is considerably simpler: the Semantic Trainer requires an expensive search of the Google Ngrams database for additional terms (that might also turn out to be unrelated visually), while we mine these terms from the actually-used tags. In addition, the quality filter (*qual-f*) performs best, on average. However, there are some odd, very low, numbers for certain classes and strategies: the simple search, although it uses no strategy at all to cull noise and false positives, does very well, except for the *tv monitor* class. Following the methodology of the Semantic Trainer, unlike all other classes where the search keyword is the name of the class, the keyword for this class is the single word "tv". This creates a problem specifically heightened in the Flickr community, as the users have appropriated the acronym "tv" for something completely different from the television device. Hence, the simple search performance is terrible, while some of the Semantic Trainer strategies and some of our proposed methods are able to correctly focus on the monitor device. The *sofa* class also results very difficult to handle. As shown in Figure 1, this particular object is often an accessory to other attractions. In the last set of experiments, we will discuss some workarounds for these problems.

Of significant interest for the practical usefulness of our approach is how well the training datasets generalize beyond the insights gathered on Pascal VOC. One way to gain an approximate idea is by performing cross-dataset evaluations between different benchmark datasets, and comparing the relative performance of our training sets. Following [19], we set out to explore cross-dataset generalization, meaning to perform cross-dataset evaluations between different benchmark datasets, and comparing the relative performance of our training sets. In particular, we analyze the behavior of the *person* class, which is of particular interest for many reasons, spanning from multimedia to social robotics, from surveillance to human computer interaction. For each class, we perform 10 randomized experiments with 200 positive and 400 negative samples split into 50% for training, 25% for cross-validation, and 25% for testing. As a source of negative samples, we use the "other" classes of Pascal VOC.

From the results in Table 2, our noun filter gives the best result on average among all the approaches of automatic training set generation, having the top scores when considering the Caltech-256 [30] and Pascal VOC. It is also worth noting that on the Pascal VOC dataset all our filters give the best performance. Finally, it is encouraging to see that our best score is comparable to what is obtained by ImageNet.

Table 2. Cross-dataset generalization on the *person* class: rows and columns identify training and testing datasets, respectively. The last column averages all the results in a given row, where this average excludes matching sources for the public databases.

Train on:	**Test on**:				**Mean**
	ImageNet	**Graz**	**Caltech-256**	**Pascal VOC**	**others**
Pascal VOC	95.10	92.22	97.04	94.71	94.77
Graz [31]	92.10	99.46	94.32	88.06	93.48
Caltech-256 [32]	96.44	90.42	99.33	92.87	94.77
ImageNet	99.14	93.59	97.88	92.39	95.75
ccom [19]	97.61	97.76	96.07	88.01	94.86
fcom [19]	95.52	94.90	94.79	87.54	93.12
freq-f	95.72	95.72	95.03	88.22	93.68
qual-f	96.48	90.53	96.22	89.85	93.27
keyw-f	96.22	94.58	96.51	90.34	94.41
noun-f	97.21	94.50	98.00	91.28	95.25

4.2. Comparison against OPTIMOL

A relatively harder comparison is with OPTIMOL [18], which analyzes the content of the images: in fact, our approach is agnostic towards the visual information, working only on textual data. For the sake of comparison we adopt the same experimental protocol, considering a selection of classes of the Caltech-101 [1,33], and generating the same number of training images. As for the testing set, we consider all the images provided by the Caltech-101. As for the features, we extract 128-dimensions dense SIFT, quantizing them into a 100-visual word dictionary by applying *k*-means clustering provided by the Vlfeat library (v0.9.20) [34]. We then use these histograms to train a linear SVM (with Liblinear v1.94 [35]) for each class and to perform object classification. In Table 3, the classification results are reported, showing that surprisingly all of our approaches work better than OPTIMOL, with the frequency filter outperforming it by more than 14%.

Table 3. Comparison against OPTIMOL. Our framework improves on it by more than 14%.

	OPTIMOL	**Our Framework**			
		freq-f	*qual-f*	*keyw-f*	*noun-f*
aeroplane	76.00	84.07	69.10	79.21	79.87
car	94.50	95.20	94.98	95.11	94.84
face	82.90	83.44	83.32	78.40	90.70
guitar	60.40	97.14	96.99	98.09	97.03
leopard	89.00	92.24	95.49	91.80	92.21
motorbike	67.30	75.83	63.67	71.77	69.03
watch	53.60	94.66	95.98	90.45	89.58
Mean AP	74.81	88.94	85.65	86.40	87.61

4.3. Entropy-Based Selection

This section describes a set of experiments performed quite some time after the ones shown in the previous sections. Although we made some minor changes to the way we collect information, a major disadvantage in using social media platforms like Flickr is the impermanence of their content: in part because the community of users continuously updates it, and in part because the company might change the inner workings of the search engine to correct defects or improve performances. Thus, there is no guarantee that the same algorithms return the same results.

In these experiments, we first establish a new baseline dataset that corrects two of the problems of the simple search in Section 4.1: first, we change our approach towards the *tv monitor* class (or any other multi-word class), by treating both its keywords as tags to be required in an image. This is possible because the constraint present for the Semantic Trainer is artificial in our framework, and we are not limited to bi-grams as keywords, but we can add any number of additional tags. Secondly, we observed that some of the classes often have a sizeable intersection in their datasets; for example, *cat* and *sofa* generate, separately, many images with both objects, or *person* with many of the other classes. Since our framework does not check the contents of the images, the next best thing we can do is require the search engine to exclude images tagged with the other keywords when querying for a certain class.

Thus, we repeated the Pascal VOC 2012 comparative experiments using these settings. Other small fixes were made in how the class dictionaries are built: we used 1000 tagged images instead of 500 to build the initial dictionaries, for additional precision in the frequencies, although no substantial changes in the keywords; since Flickr is a major community of photography enthusiasts, there are often tags specifying the camera device used or the photoshoot setup: some of these words (like "canon", "eos", "dslr", *etc*.) were pruned from the raw dictionaries. It is feasible, in a future work, to automatically detect such biases from a global corpus dictionary, where such biases would show up as tags highly used independently from the image subject.

In Table 4, the "old Flickr" column reports the same simple search numbers of Table 1, while the "new Flickr" column shows the new situation exclusively due to changes in the Flickr content and search engine (except for the *tv monitor* multi-word fix). Classes like *cow* and *motorbike* have dropped considerably, while *person* has improved by a large margin. Unfortunately, these changes are out of our control, and they ultimately affect our framework's performance. However, once the new baseline was established, where other classes' keywords are constrained to be excluded, we can see that both the frequency filter and the entropy-based selection strategy provide definite improvements, with the latter outperforming the former on average and on more than half of the classes. We chose to collect $k = 10$ keywords for both methods.

In Figure 2, we show the gains graphically to better highlight the changes: the only considerable drop is in the *sheep* class, while there are several sizeable improvements in the *motorbike*, *cow*, and *person* classes. In Table 5, we show all the tags selected for the entropy-based experiment: below each tag is the corresponding value h_i of the (relative) entropy as calculated in Section 3. The tag "monochrome", which is present in several classes, is another word tied to photography setups that jumps out only from an overview. In Figure 3, we show the relative importance of the selected tags for the four classes mentioned above. The pie charts show the relative shares of the total entropy; that is, $h_i / \sum_i h_i$, which are used to calculate the number of images each expanded query contributes to the final class dataset. In Table 6, we show some representative images in the datasets assembled for the *motorbike* and *sofa* classes. The former set of images shows a good selection of correct images, and a few examples of misleading ones. The latter highlights problems: some of the images have the wrong focus, and the selected tags themselves are unhelpful. In both sets, there are near-similar images probably coming from the same source: this can be either an advantage when the source can be trusted to provide useful images, or an hindrance, as it will multiply the number of wrong images.

Table 4. Results of the new experiments: the old and new Flickr simple search are reported for reference; in the baseline, we exclude the keywords of other classes in the simple search; the entropy-based selection strategy is compared to the frequency filter (with changes calculated against the baseline; bold font indicates the best for each class).

Classes	Old Flickr	New Flickr	Baseline	*freq-f*	(*Difference*)	Entropy	(*Difference*)
aeroplane	97.7	96.7	**96.8**	96.5	(−0.3)	96.8	(0)
bicycle	82.8	79.4	79.8	79.9	(+0.1)	**81.7**	(+1.9)
bird	90.7	88.6	89.3	90.4	(+1.1)	**90.9**	(+1.6)
boat	88.7	84.3	84.8	**85.4**	(+0.6)	85.2	(+0.4)
bottle	57.3	58.9	57.6	**59.0**	(+1.4)	58.8	(+1.2)
bus	93.8	93.7	**93.3**	92.9	(−0.4)	92.6	(−0.7)
car	72.6	71.2	71.6	**73.0**	(+1.4)	72.5	(+0.9)
cat	91.5	92.9	94.2	94.0	(−0.2)	**94.7**	(+0.5)
chair	70.3	65.4	69.3	67.7	(−1.6)	**70.3**	(+1.0)
cow	79.0	65.8	66.0	70.0	(+4.0)	**70.4**	(+4.4)
dog	88.9	86.7	87.8	88.0	(+0.2)	**88.1**	(+0.3)
horse	85.1	80.9	82.5	**84.9**	(+2.4)	84.0	(+1.5)
motorbike	89.1	74.5	76.2	78.5	(+2.3)	**83.1**	(+6.9)
person	60.4	80.7	80.6	82.9	(+2.3)	**84.4**	(+3.8)
sheep	84.9	81.1	**82.0**	75.3	(−6.7)	77.0	(−5.0)
sofa	58.0	50.2	54.8	55.6	(+0.8)	**56.8**	(+2.0)
train	92.8	92.2	92.0	**93.3**	(+1.3)	93.1	(+1.1)
tv monitor	25.0	79.4	79.5	79.8	(+0.3)	**80.6**	(+1.1)
Mean AP	78.3	79.0	79.9	80.4	(+0.5)	**81.2**	(+1.3)

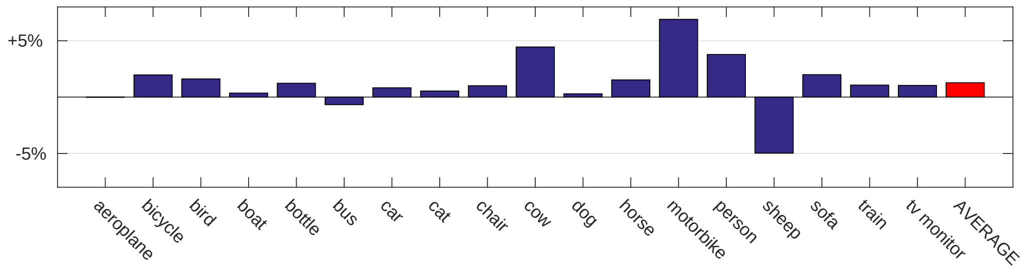


Figure 2. Graph of the gains for the entropy-based selection strategy against the baseline.

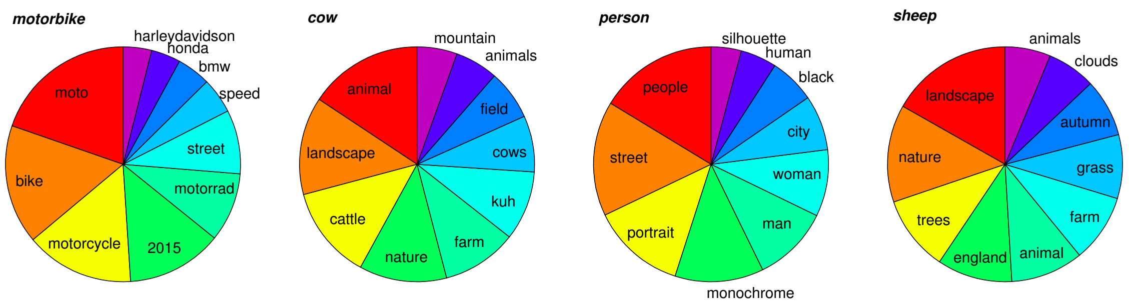
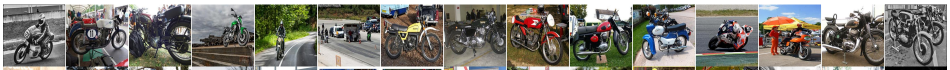
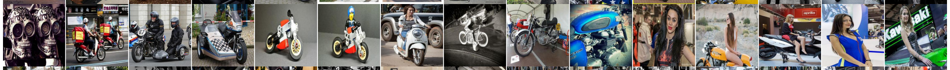
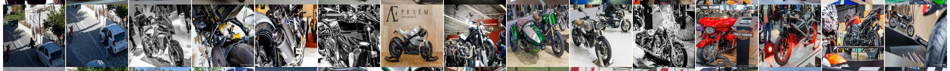
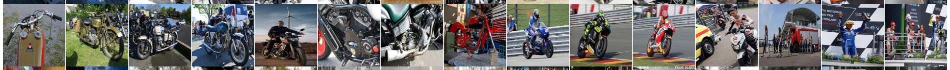
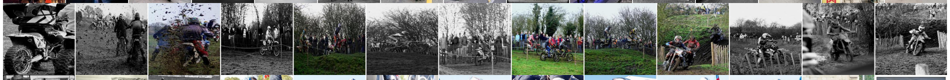
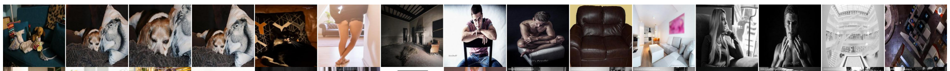
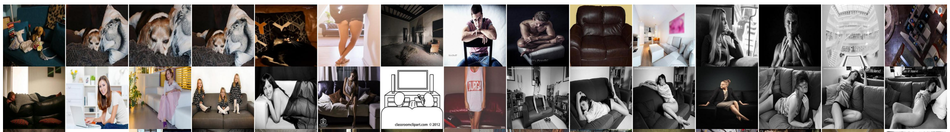
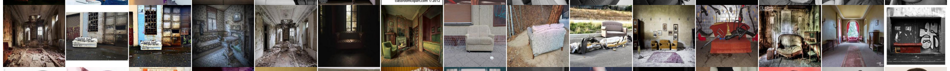
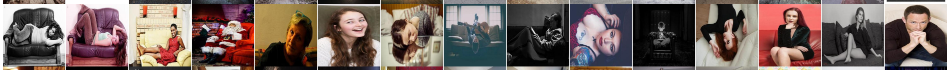
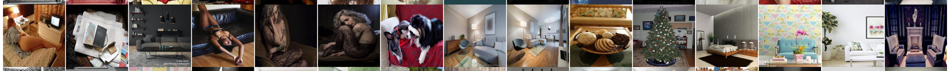
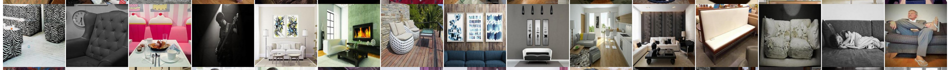
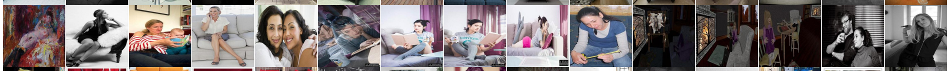
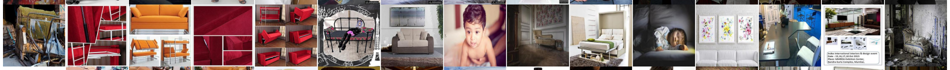
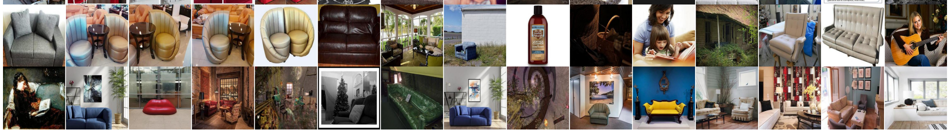
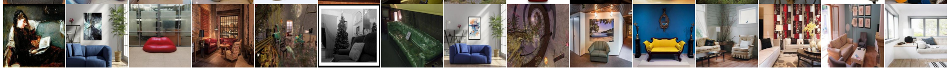


Figure 3. Charts of the relative importance of the selected tags based on the entropy value h_i for the classes *motorbike*, *cow*, *person*, and *sheep*.

Table 5. All the tags selected in the entropy-based strategy, with the corresponding (relative) entropy values below each tag.

Classes	**Selected Tags**									
aeroplane	plane	airport	aviation	airplane	jet	flight	boeing	aircraft	raf	planespotting
h_i	1.00	0.97	0.84	0.76	0.70	0.61	0.57	0.51	0.38	0.31
bicycle	bike	street	city	people	europe	monochrome	fahrrad	cycle	man	shadow
h_i	1.00	0.90	0.57	0.55	0.47	0.37	0.35	0.31	0.29	0.26
bird	nature	wildlife	animal	birds	outdoor	vogel	water	oiseau	fauna	ngc
h_i	1.00	0.84	0.75	0.67	0.55	0.47	0.45	0.40	0.32	0.25
boat	water	sea	landscape	sky	sunset	reflection	clouds	summer	ship	blue
h_i	1.00	0.93	0.81	0.77	0.74	0.62	0.58	0.51	0.46	0.39
bottle	glass	green	wine	water	drink	beer	stilllife	blue	macro	red
h_i	0.78	0.53	0.50	0.45	0.43	0.42	0.39	0.36	0.33	0.32
bus	london	street	city	transport	night	people	uk	road	travel	buses
h_i	0.78	0.73	0.60	0.49	0.41	0.40	0.37	0.34	0.29	0.28
car	auto	vintage	street	cars	red	automobile	old	night	urban	classic
h_i	0.73	0.67	0.64	0.58	0.50	0.47	0.43	0.41	0.37	0.35
cat	animal	pet	kitten	portrait	chat	kitty	cats	feline	eyes	nature
h_i	0.97	0.68	0.66	0.60	0.55	0.46	0.41	0.36	0.33	0.27
chair	abandoned	window	table	urban	white	art	red	shadow	old	beach
h_i	0.73	0.56	0.55	0.47	0.42	0.34	0.34	0.32	0.30	0.29
cow	animal	landscape	cattle	nature	farm	kuh	cows	field	animals	mountain
h_i	0.80	0.69	0.66	0.61	0.53	0.49	0.39	0.35	0.30	0.28
dog	animal	pet	portrait	white	nature	hund	street	beach	puppy	dogs
h_i	0.87	0.71	0.59	0.46	0.44	0.38	0.38	0.35	0.34	0.30
horse	horses	animal	cheval	landscape	nature	white	equine	sky	sunset	outdoor
h_i	0.93	0.66	0.57	0.47	0.44	0.35	0.32	0.30	0.28	0.27
motorbike	moto	bike	motorcycle	2015	motorrad	street	speed	bmw	honda	harleydavidson
h_i	0.99	0.82	0.75	0.66	0.48	0.44	0.25	0.23	0.20	0.20
person	people	street	portrait	monochrome	man	woman	city	black	human	silhouette
h_i	1.00	0.97	0.79	0.75	0.64	0.56	0.46	0.38	0.31	0.25
sheep	landscape	nature	trees	england	animal	farm	grass	autumn	clouds	animals
h_i	0.85	0.68	0.52	0.52	0.50	0.47	0.44	0.40	0.34	0.32
sofa	couch	girl	abandoned	portrait	home	white	woman	bed	furniture	interior
h_i	0.75	0.59	0.50	0.46	0.38	0.33	0.29	0.27	0.25	0.24
train	railway	station	railroad	city	rail	locomotive	travel	street	trains	monochrome
h_i	0.85	0.80	0.63	0.59	0.49	0.42	0.38	0.36	0.33	0.30
tv monitor	television	video	screen	3d	electronics	art	film	computer	germany	lcd
h_i	0.89	0.79	0.55	0.48	0.40	0.34	0.29	0.27	0.24	0.21

Table 6. Representative images in the datasets assembled for the classes *motorbike* and *sofa*. Each row shows up to 15 images contributed by the corresponding expanded query with the selected tag. The tags are in the order they were selected by the entropy-based strategy.

Tag	Representative Images
motorbike	
moto	
bike	
motorcycle	
2015	
motorrad	
street	
speed	
bmw	
honda	
harleydavidson	
sofa	
couch	
girl	
abandoned	
portrait	
home	
white	
woman	
bed	
furniture	
interior	

5. Conclusions

The exploitation of data from social media is an ongoing and very successful trend in machine learning, one that is bound to bear fruit for many years to come as the wealth of content increases and more information is extracted and made useful. Automatic generation of training sets for classifiers is gradually becoming a viable path towards the rapid utilization of this data, especially given the requests for classifiers embedded into portable devices like smartphones. Take for example the popular applet Shazam, which recently introduced visual recognition (see [36]). In this work, we have shown that textual tags associated with images from social media platforms, even if noisy, represent an important source of information expressive enough to generate image datasets that compare favorably with man-made repositories such as Pascal VOC and ImageNet. Future improvements include associating metadata with textual tags, and incorporating analyses of the visual content in the images, trying to understand the relationship among visual features and textual features. A framework that connects visual, textual, and metadata information is what we envisage in the future for creating a theory of *visual knowledge* able to inform intelligent systems about the world around us.

Supplementary Materials: The representative images of the 18 assembled datasets described in the paper are available online at www.mdpi.com/1099-4300/18/4/130/s1.

Acknowledgments: Dong Seon Cheng was supported by the Hankuk University of Foreign Studies Research Fund of 2015. Both authors have read and approved the final manuscript.

Author Contributions: Sami Abduljalil Abdulhak wrote the initial draft describing the state of the art, the filters and the first two sets of experiments, Dong Seon Cheng designed the entropy-based selection strategy, performed the last set of experiments, introduced the overall framework perspective, and revised the entire article. Both authors have read and approved the final manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Fei-Fei, L.; Fergus, R.; Perona, P. A Bayesian approach to unsupervised one-shot learning of object categories. In Proceedings of the 9th IEEE International Conference on Computer Vision, Nice, France, 13–16 October 2003; pp. 1134–1141.
2. Crowston, K. Amazon Mechanical Turk: A Research Tool for Organizations and Information Systems Scholars. In *Shaping the Future of ICT Research. Methods and Approaches*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 210–221.
3. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009), Miami, FL, USA, 20–25 June 2009; pp. 248–255.
4. Miller, G.A. WordNet: A lexical database for English. *Commun. ACM* **1995**, *38*, 39–41.
5. Flickr. Available online: http://www.flickr.com (accessed on 7 April 2016).
6. Ames, M.; Naaman, M. Why we tag: Motivations for annotation in mobile and online media. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, San Jose, CA, USA, 30 April–3 May 2007; pp. 971–980.
7. Petz, G.; Karpowicz, M.; Fürschuß, H.; Auinger, A.; Stříteský, V.; Holzinger, A. Computational approaches for mining user's opinions on the Web 2.0. *Inf. Process. Manag.* **2014**, *50*, 899–908.
8. Kennedy, L.S.; Chang, S.F.; Kozintsev, I.V. To search or to label?: Predicting the performance of search-based automatic image classifiers. In Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval (MIR '06), Santa Barbara, CA, USA, 23–27 October 2006; pp. 249–258.
9. Mandala, R.; Tokunaga, T.; Tanaka, H. Combining multiple evidence from different types of thesaurus for query expansion. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Berkeley, CA, USA, 15–19 August 1999; pp. 191–197.
10. Everingham, M.; van Gool, L.; Williams, C.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338.
11. Visual Object Classes Challenge 2012. Available online: http://host.robots.ox.ac.uk/pascal/VOC/voc2012/index.html (accessed on 7 April 2016).
12. Imagenet Large Scale Visual Recognition Challenge 2012. Available online: http://www.image-net.org/challenges/LSVRC/2012/index (accessed on 7 April 2016).
13. Li, L.J.; Fei-Fei, L. Optimol: Automatic online picture collection via incremental model learning. *Int. J. Comput. Vis.* **2010**, *88*, 147–168.

14. Wang, G.; Hoiem, D.; Forsyth, D. Learning image similarity from Flickr groups using stochastic intersection kernel machines. In Proceedings of the 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 428–435.
15. Fergus, R.; Fei-Fei, L.; Perona, P.; Zisserman, A. Learning Object Categories from Google's Image Search. In Proceedings of the 10th International Conference on Computer Vision, Beijing, China, 17–21 October 2005; Volume 2, pp. 1816–1823.
16. Chen, X.; Shrivastava, A.; Gupta A. NEIL: Extracting Visual Knowledge from Web Data. In Proceedings of the 14th International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 1409–1416.
17. Divvala, S.K.; Farhadi, A.; Guestrin, C. Learning Everything about Anything: Webly-Supervised Visual Concept Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14), Columbus, OH, USA, 23–28 June 2014; pp. 3270–3277.
18. Li, L.J.; Wang, G.; Fei-Fei, L. OPTIMOL: Automatic Online Picture collecTion via Incremental Model Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '07), Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8.
19. Cheng, D.S.; Setti, F.; Zeni, N.; Ferrario, R.; Cristani, M. Semantically-driven automatic creation of training sets for object recognition. *Comput. Vis. Image Underst.* **2015**, *131*, 56–71.
20. WordNet. Available online: http://wordnet.princeton.edu (accessed on 7 April 2016).
21. Sun, A.; Bhowmick, S.S. Image Tag Clarity: In Search of Visual-Representative Tags for Social Images. In Proceedings of the 1st SIGMM Workshop on Social Media (WSM '09), Beijing, China, 19–24 October 2009; pp. 19–26.
22. Spain, M.; Perona, P. Some Objects Are More Equal than Others: Measuring and Predicting Importance. In Proceedings of the 10th European Conference on Computer Vision: Part I (ECCV '08), Marseille, France, 12–18 October 2008; Springer: Berlin/Heidelberg, Germany, 2008; pp. 523–536.
23. Weinberger, K.Q.; Slaney, M.; van Zwol, R. Resolving tag ambiguity. In Proceeding of the 16th ACM International Conference on Multimedia, ACM, MM '08, Vancouver, BC, Canada, October 26–31 2008; pp. 111–120.
24. Shepitsen, A.; Gemmell, J.; Mobasher, B.; Burke, R. Personalized Recommendation in Social Tagging Systems Using Hierarchical Clustering. In Proceedings of the 2008 ACM Conference on Recommender Systems (RecSys '08), Lausanne, Switzerland, 23–25 October 2008; pp. 259–266.
25. Hassan-Montero, Y.; Herrero-Solana, V. Improving Tag-Clouds as Visual Information Retrieval Interfaces. In Proceedings of the International Conference on Multidisciplinary Information Sciences and Technologies (InSciT2006), Mérida, Spain, 25–28 October 2006.
26. Ogden, C.K. *Basic English: A General Introduction with Rules and Grammar*; Kegan Paul: London, UK, 1932.
27. Holzinger, A.; Hörtenhuber, M.; Mayer, C.; Bachler, M.; Wassertheurer, S.; Pinho, A.J.; Koslicki, D. On entropy-based data mining. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*; Holzinger, A., Jurisica, I., Eds.; Springer: Berlin/Heidelberg, 2014; pp 209–226.
28. Vedaldi, A.; Lenc, K. MatConvNet: Convolutional Neural Networks for MATLAB. In Proceedings of the 23rd ACM International Conference on Multimedia (MM '15), Brisbane, Australia, 26–30 October 2015; pp. 689–692.
29. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25*; Pereira, F., Burges, C., Bottou, L., Weinberger, K., Eds.; Curran Associates Inc.: New York, NY, USA, 2012; pp. 1097–1105.
30. Caltech 256. Available online: http://www.vision.caltech.edu/Image_Datasets/Caltech256/ (accessed on 7 April 2016).
31. Opelt, A.; Pinz, A.; Fussenegger, M.; Auer, P. Generic object recognition with boosting. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 416–431.
32. Griffin, G.; Holub, A.; Perona, P. Caltech-256 Object Category Dataset. Available online: http://authors.library.caltech.edu/7694/ (accessed on 7 April 2016).
33. Caltech 101. Available online: http://www.vision.caltech.edu/Image_Datasets/Caltech101/ (accessed on 7 April 2016).
34. Vedaldi, A.; Fulkerson, B. Vlfeat: An Open and Portable Library of Computer Vision Algorithms. In Proceedings of the 18th ACM International Conference on Multimedia (MM '10), Firenze, Italy, 25–29 October 2010; pp. 1469–1472.

35. Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; Lin, C.-J. LIBLINEAR: A library for large linear classification. *J. Mach. Learn. Res.* **2008**, *9*, 1871–1874.
36. Digital Trends. Available online: http://www.digitaltrends.com/mobile/shazam-music-app-visual-recognition/ (accessed on 7 April 2016).