Content-based visual search learned from social media

Li, X.

Citation for published version (APA):
Li, X. (2012). Content-based visual search learned from social media.
Tag Relevance Fusion for Social Image Search

*How to fuse tag relevance estimates?* In this chapter we develop early and late fusion schemes from generic multimedia analysis in the new context of tag relevance estimation. We systematically study the characteristics and performance of early and late tag relevance fusion. Experiments on a large present-day benchmark show that tag relevance fusion leads to better image search. Moreover, we find that the unsupervised fusion methods are practically as effective as the supervised alternatives, but without the need of any training efforts.

---

*A preliminary version of this work received the Best Paper Award from the ACM International Conference on Image and Video Retrieval 2010 [65]. Submitted [67].*
3.1 Introduction

Searching for the increasing amounts of varied and dynamically changing images on the social web is important. A number of applications are grounded on social image search, such as landmark visualization [55], visual query suggestion [147], training data acquisition [115], photo-based question answering [144], and photo-based advertisements [70], to name a few. Given that social images are often described by user-contributed tags, one might expect tag-based retrieval to be a natural and good starting point for image search. Compared to content-based search [27], tag-based search potentially bypasses the semantic gap problem, and its scalability has been verified by decades of text retrieval research [7]. However, due to varied reasons, including batch tagging and the diversity of user knowledge, social tagging is known to be ambiguous, subjective, and inaccurate [81]. Moreover, since individual tags are used only once per image in the social tagging paradigm, relevant and irrelevant tags for a specific image are not separable by tag statistics alone. Estimating the relevance of social tags with respect to the images they are describing is essential for generic image search.

For tag relevance estimation, quite a few methods have been proposed. For example, Liu et al. [75] utilize a random walk model to rank tags in terms of their relevance to the visual content. Chen et al. [20] train a Support Vector Machine classifier per tag. Given an image and its social tags, Zhu et al. [151] measure the relevance of a specific tag in terms of its semantic similarity to the other tags. We proposed a neighbor voting algorithm which exploits tagging redundancies among multiple users [64]. By using learned tag relevance value as a new ranking criterion, better image search results are obtained, when compared to image search using original tags.

Positioned in a deluge of social data, however, tag relevance estimation is challenging. Visual concepts for example “boat” or “garden” vary significantly in terms of their visual appearance and visual context. A single measurement of tag relevance as proposed in previous work is limited to tackle such large variations, resulting in sub-optimal image search. For large-scale data, we consider fusing multiple tag relevance estimates as an important extension to methods for tag relevance estimation.

In a broad sense of tag relevance fusion, we shall consider multiple sources of evidence such as visual content [20, 64, 75], tag co-occurrence [12, 151], social notes [89, 98], and personal tagging history [60]. Among them, visual content is the only objective piece of evidence. Hence, in this work we focus on fusing tag relevance derived from visual content analysis.

It is becoming increasingly clear that no single feature can represent the visual content completely [38, 78, 126]. Global features are suited for capturing the gist of scenes [88], while local features better depict properties of objects [122]. As shown previously in content-based image search [117, 127], image annotation [41, 80], and video concept detection [128, 137], fusing multiple visual features is beneficial. The question arises what fusion methods are suited in the new context of tag relevance estimation.
Before answering the question, we think over the main characteristics of social data: large-scale, miscellaneous, and dynamic. These characteristics imply that a fused tag relevance measurement should favorably exploit and aggregate the large amount of miscellaneous information scattered on the social web. Moreover, lightweight fusion methods are preferred to keep pace with the rapid development of social data. To establish our study, we develop the notion of early and late fusion from Snoek et al. [110] in the new context, and define

Definition 1 (Early Tag Relevance Fusion). Fusion schemes that integrate individual features before estimating social tag relevance scores.

Definition 2 (Late Tag Relevance Fusion). Fusion schemes that first use individual features to estimate social tag relevance scores separately, and then integrate the scores.

The main contributions of this chapter are as follows.

• We propose tag relevance fusion as an extension of tag relevance estimation.

• Using the neighbor voting algorithm as a base tag relevance estimator [64], we present a systematic study on early and late tag relevance fusion. We extend the base estimator for both early and late fusion. Our previous work [65], which discusses late tag relevance fusion only, is a special case of this work.

• Experiments on a large benchmark show that tag relevance fusion leads to better image search.

• This study provides a practical mechanism to exploit diverse visual features in interpreting tag relevance for social image search.

The problem we study lies at the crossroads of social tag relevance estimation and visual fusion. So next we present a literature review of both areas.

3.2 Related Work

3.2.1 Social Tag Relevance Estimation

A number of methods have been proposed to attack the tag relevance estimation problem [20, 56, 64, 74, 75, 112, 138, 149, 151]. We structure these methods in terms of the main rationale they use to estimate tag relevance. We summarize the rationale into the following three aspects: visual consistency [56, 64, 75, 112], semantic consistency [20, 151], and visual-semantic consistency [74, 149]. Given two images labeled with the same tag, the visual consistency based methods conjecture that if one image is visually closer to images labeled with the tag than the other image, then the former image is more relevant to the tag. Liu et al. [75] use a random walk model to find such visually close images. Sun and Bhowmick [112] exploit visual consistency to quantify
a tag’s relevance to the visual content. We proposed a neighbor voting algorithm which infers the relevance of a tag with respect to an image by counting its visual neighbors labeled with that tag [64]. Lee et al. [56] first identify tags which are suited for describing the visual content by a dictionary lookup. Later, they apply the neighbor voting algorithm to the identified tags. Zhu et al. [151] investigate semantic consistency, measuring the relevance of a tag to an image in terms of its semantic similarity to the other tags assigned to the image, ignoring the visual content of the image itself. Chen et al. [20] assume that photos uploaded by the same user within a short time span form a semantically related group. They train SVM models for individual tags, and use the models to estimate image tag relevance within the photo group. To jointly exploit visual and semantic consistency, Liu et al. [74] perceive tag relevance estimation as a semi-supervised multi-label learning problem, while Zhu et al. [149] formulate the problem as decomposing an image tag co-occurrence matrix. In all the above methods, only a single feature is considered. We hypothesize that fusing multiple tag relevance estimates driven by diverse features could improve such methods.

### 3.2.2 Visual Fusion

Snoek et al. [110] classify fusion methods into two groups: early fusion and late fusion. We follow their taxonomy to organize our literature review on visual fusion. In early fusion, a straightforward method is to concatenate individual features to form a new single feature [3, 110]. As feature dimensionality increases, the method suffers from the curse of dimensionality [92]. Another disadvantage of the method is the difficulty to combine features into a common representation [110]. Instead of feature concatenation, another method is to combine visual similarities of the individual features [41, 80, 128]. In the context of image annotation, Makadia et al. [80] and Guillaumin et al. [41] linearly combine multiple visual similarities. In a video concept detection context, Wang et al. [128] also choose linear fusion to combine similarity graphs defined by different features. In late fusion, models are obtained separately on the individual features and their output is later combined [126, 137]. Wu et al. [137] first train base classifiers using distinct features. Then, they view the output of the base classifiers as a new feature to obtain a final classifier. Wang et al. [126] adaptively combine the base classifiers in a boosting framework. To the best of our knowledge, visual fusion in the tag relevance estimation context has not been well explored.

### 3.3 Base Tag Relevance Estimators

For a valid comparison between early and late fusion, we should choose the same base tag relevance estimators for both fusion schemes. Thus, before delving into the discussion about tag relevance fusion and its solutions, we first make our choice of base estimators. For the ease of consistent description, we use $x$ to denote an image,
and $w$ for a social tag. Let $g(x, w)$ be a base tag relevance function whose output is a confidence score of a tag being relevant to an image. Further, let $S$ be a large set of social-tagged images, and $S_w$ the set of images labeled with $w$, $S_w \subset S$.

A base estimator should be data-driven and favorably exploit the large amount of social data. Moreover, it should be generic enough to adapt to both early and late fusion. In that regard, we choose the neighbor voting algorithm proposed in our previous work [64]. A recent study by Sun et al. [113] indicates that this algorithm is the most effective for tag relevance estimation. In order to find visual neighbors from $S$ for a given image $x$, we use $f(x)$ to represent a specific visual feature vector, and $S_{x,f,k}$ as the $k$ nearest visual neighbors of $x$, measured in terms of $f$. We also have to specify a distance function for each given feature. The optimal distance varies in terms of tasks [41, 94]. Here for all features, we choose the Euclidean distance for its widespread use in the literature. In the neighbor voting algorithm, $g(x, w)$ is computed as

$$g(x, w) = \frac{|S_{x,f,k} \cap S_w|}{k} - \frac{|S_w|}{|S|},$$

where $|\cdot|$ is the cardinality of a set. We see from (3.1) that more neighbor images labeled with the tag induces larger tag relevance scores, while common tags which have high frequency and thus low descriptive power are suppressed by the second term. We conceptualize early and late tag relevance fusion with diagrams in Fig. 6.1, and elucidate them next.

### 3.4 Tag Relevance Fusion

#### 3.4.1 Problem Formalization

From an information fusion perspective [14], diversity in base tag relevance estimators is important for effective fusion. We can generate multiple tag relevance estimators by varying the visual feature $f$, the number of neighbors $k$, or both. For a given feature, as the larger set of visual neighbors always includes the smaller set of visual neighbors, the parameter $k$ has a limited impact on the diversity. Hence, we fix $k$ and diversify the base estimators by using diverse visual features. Let $F = \{f_1, \ldots, f_m\}$ be a set of such features. We use $g_i(x, w)$ to denote a base estimator specified by feature $f_i$, $i = 1, \ldots, m$. For the two fusion schemes defined in Section 5.1, we use $G^e(x, w)$ to denote a fused tag relevance estimator obtained by early fusion, and $G^l(x, w)$ to denote a late fused estimator.

Since linear fusion is a well accepted choice for visual fusion as described in Section 3.2.2, we follow this convention for tag relevance fusion. For early fusion, we aim for a better neighbor set by combining visual similarities defined by the $m$ features. Concretely, given two images $x$ and $x'$, let $\text{sim}_i(x, x')$ be a visual similarity defined
Chapter 3. Tag Relevance Fusion for Social Image Search

**Figure 3.1:** The two proposed tag relevance fusion schemes. We use the neighbor voting algorithm [64] to realize base tag relevance estimators. Given an image \( x \), different textured backgrounds indicate its visual neighbors obtained by diverse features. In Early Tag Relevance Fusion (a), we fuse multiple visual neighbor sets (denoted by \( \downarrow \)) to obtain a better neighbor set for tag relevance estimation. In Late Tag Relevance Fusion (b), we fuse multiple tag relevance estimates (denoted by \( \triangledown \)).

by feature \( f_i \). We define the fused visual similarity as

\[
sim_\Lambda(x, x') = \sum_{i=1}^{m} \lambda_i \cdot \sim_{i}(x, x'),
\]

where \( \lambda_i \) is a weight indicating the importance of feature \( f_i \) in the fusion process. The subscript \( \Lambda \) makes the dependence of the fused similarity on \( \{\lambda_i\} \) explicit. We choose features which are intellectually devised, so we assume that they are better than random guessing, meaning adding them is helpful for measuring the visual similarity. Hence, we constrain our solution with \( \lambda_i \geq 0 \). Since normalizing weights by dividing by their sum does not affect image ranking, any linear fusion with nonnegative weights can be transformed to a convex combination. So we enforce \( \sum_{i=1}^{m} \lambda_i = 1 \). Replacing the similarity used in (3.1) by the fused similarity (3.2) leads to the early fused tag relevance function:

\[
G^\Lambda_{x}(x, w) = \frac{|S_{x,\Lambda,k} \cap S_{w}|}{|S_{w}|},
\]

where \( S_{x,\Lambda,k} \) denotes the \( k \) nearest neighbors obtained by \( \sim_{\Lambda}(x, x') \).
In a similar fashion, we define our linear late fused tag relevance function:

\[ G^l_\Lambda(x, w) = \sum_{i=1}^{m} \lambda_i \cdot g_i(x, w). \quad (3.4) \]

In order to study the statistical properties of \( G^e_\Lambda(x, w) \) and \( G^l_\Lambda(x, w) \), we extend the derivation of a single estimator in our earlier study [64] to the new context. The effectiveness of an estimator \( g(x, w) \) depends on the accuracy of social tagging and visual neighbor search [64]. To describe the social tagging accuracy, we divide the social image set \( S \) into two disjoint subsets, \( S^+ \) and \( S^- \), where images in \( S^+ \) are relevant to \( w \) and images in \( S^- \) are irrelevant to \( w \). Let \( Q^+ \) be the probability of correct tagging, i.e., an image randomly sampled from \( S^+ \) is labeled with \( w \), and \( Q^- \) be the probability of incorrect tagging, i.e., an image randomly sampled from \( S^- \) is labeled with \( w \). We have established in our previous work [64] that in the social web the probability of correct tagging is larger than the probability of incorrect tagging, thus

\[ Q^+ - Q^- > 0. \quad (3.5) \]

Given an image relevant to \( w \), the accuracy of neighbor search by a specific feature \( f \) is the percentage of neighbors which are also relevant to \( w \). To describe the neighbor search accuracy, we consider random sampling as a baseline. Let \( P^+ \) be the probability that an image randomly sampled from \( S^+ \) is relevant to \( w \). We introduce an offset variable \( \varepsilon_{x, w, f} \) to reflect the relative accuracy compared to random sampling. In particular, we use \( (P^+ + \varepsilon_{x, w, f}) \) to represent the probability that an image randomly sampled from the neighbor set \( S_{x, f, k} \) is relevant to \( w \). We can re-express \( g(x, w) \) as

\[ g(x, w) = \varepsilon_{x, w, f} \cdot (Q^+ - Q^-). \quad (3.6) \]

For the derivation of (3.6), we refer to the paper [64]. Consequently, for any image \( x^+ \) relevant to \( w \) and any image \( x^- \) irrelevant to \( w \), if

\[ \varepsilon_{x^+, w, f} - \varepsilon_{x^-, w, f} > 0, \]

we will have \( g(x^+, w) - g(x^-, w) > 0 \), meaning an ideal tag relevance estimator which ranks all relevant images in front of all irrelevant images.

To describe the neighbor search accuracy of the fused similarity (3.2), we use \( \varepsilon_{x, w, \Lambda} \) to denote the corresponding offset variable. Substituting \( \varepsilon_{x^+, w, \Lambda} \) for \( \varepsilon_{x^+, w, f} \) in (3.6), we re-write \( G^e_\Lambda(x, w) \) as

\[ G^e_\Lambda(x, w) = \varepsilon_{x, w, \Lambda} \cdot (Q^+ - Q^-). \quad (3.8) \]

For images \( x^+ \) and \( x^- \), the difference between their tag relevance values is

\[ G^e_\Lambda(x^+, w) - G^e_\Lambda(x^-, w) = (\varepsilon_{x^+, w, \Lambda} - \varepsilon_{x^-, w, \Lambda}) \cdot (Q^+ - Q^-). \quad (3.9) \]
In the context of image annotation, Makadia et al. [80] and Guillamim et al. [41] report that combining diverse features improves the neighbor search accuracy. According to their studies, we make our assumption about early fusion:

**Assumption 1 (Early fusion).** A fused visual similarity is better than visual similarities using individual features, i.e., $\varepsilon_{x_{w+},w,A} - \varepsilon_{x_{w-},w,A} > 0$ is more likely to be valid than $\varepsilon_{x_{w+},w,f} - \varepsilon_{x_{w-},w,f} > 0$.

Due to the limitation of individual features for representing the visual content and finding the correct visual neighbors, inequality (3.7) might be violated. Given the diverse feature set, the inequality of the fused similarity is more likely to hold. Having inequality (3.5) and assumption 1, we can conclude that

$$G_{\Lambda}^l(x_{w+}, w) - G_{\Lambda}^l(x_{w-}, w) > 0$$

is more likely to be valid than

$$g(x_{w+}, w) - g(x_{w-}, w) > 0.$$  

This means the early fused estimator is better than the base estimators for ranking relevant images ahead of irrelevant images.

For late tag relevance fusion, by putting (3.6) into (3.4), we re-write $G_{\Lambda}^l(x, w)$ as

$$G_{\Lambda}^l(x, w) = \left( \sum_{i=1}^{m} \lambda_i \cdot \varepsilon_{x,w,f_i} \right) \cdot (Q_{w+} - Q_{w-}).$$

(3.10)

For images $x_{w+}$ and $x_{w-}$, the difference between their tag relevance values is

$$G_{\Lambda}^l(x_{w+}, w) - G_{\Lambda}^l(x_{w-}, w) = (Q_{w+} - Q_{w-}) \sum_{i=1}^{m} \lambda_i \cdot \left( \varepsilon_{x_{w+},w,f_i} - \varepsilon_{x_{w-},w,f_i} \right).$$

(3.11)

According to (3.11), as long as inequality (3.7) is valid for the majority of the features, we will have $G_{\Lambda}^l(x_{w+}, w) - G_{\Lambda}^l(x_{w-}, w) > 0$. We describe this by our assumption on late fusion:

**Assumption 2 (Late fusion).** For a diverse range of concepts, on average, $\sum_{i=1}^{m} \lambda_i \cdot \left( \varepsilon_{x_{w+},w,f_i} - \varepsilon_{x_{w-},w,f_i} \right) > 0$ is more likely to be valid than $\varepsilon_{x_{w+},w,f_i} - \varepsilon_{x_{w-},w,f_i} > 0$ for any individual feature $f_i$.

Using diverse features results in base tag relevance estimators complementary to each other. Combining (3.5) and assumption 2, we can conclude that $G_{\Lambda}^l(x_{w+}, w) - G_{\Lambda}^l(x_{w-}, w) > 0$ is more likely to be valid than $g(x_{w+}, w) - g(x_{w-}, w) > 0$. This means the late fused estimator is better than the base estimators for ranking relevant images in front of irrelevant images.

From assumption 1 and assumption 2 we see that the effectiveness of tag relevance estimation no longer counts on a specific feature, but on the majority of the features used. This provides a theoretical motivation for fusing multiple tag relevance estimates driven by diverse features.

As shown in (3.3) and (3.4), for both early and late fusion, we have expressed their corresponding tag relevance functions in a convex combination. It is thus possible to leverage the same optimization algorithms for determining the fusion parameters. In order to evaluate a specific $\Lambda$, we need a performance measure such as average...
3.4. Tag Relevance Fusion

precision to assess image rankings obtained by \( G_\Lambda(x, w) \), where \( G_\Lambda(x, w) \) is \( G^*_\Lambda(x, w) \) in early fusion and \( G^\prime_\Lambda(x, w) \) in late fusion. To formalize the optimization process, let \( D_w \) be a set of training images for tag \( w \), where each image is labeled with the tag by social tagging, with ground truth created by manual verification. Let \( \text{rank}(D_w; G_\Lambda) \) be a ranking of \( D_w \), obtained by sorting \( D_w \) in descending order by \( G_\Lambda(x, w) \). We use \( \pi(\text{rank}(D_w; G_\Lambda)) \) to represent a performance measure function which evaluates the ranking accuracy in terms of the ground truth. The goal of supervised fusion is to find a \( \Lambda \) such that \( \pi \) is maximized on \( D_w \), formally

\[
\Lambda^* = \arg\max_\Lambda \pi(\text{rank}(D_w; G_\Lambda)),
\]

with the convexity constraint,

\[
\Lambda = \{\lambda_1, \ldots, \lambda_m\}, \lambda_i \geq 0, \sum_{i=1}^{m} \lambda_i = 1.
\]

3.4.2 Fusion Methods

It is apparent that for early (late) fusion, better features (estimators) should have larger weights. The intuition helps us devise learning algorithms which exploit the performance measure as feedback to exclude many suboptimal weights, without the need to evaluate them. Moreover, performance measures widely used in the literature such as average precision are not differentiable. So we have to choose an algorithm which does not require gradient computation. Bearing these in mind, we choose the coordinate ascent based optimization technique, exploited by Metzler and Croft [82] in the domain of document retrieval. Here, we apply the technique to supervised early and late tag relevance fusion.

**Supervised Fusion: Coord-Ascent.** The Coordinate Ascent method iteratively solves (3.12) by optimizing merely one parameter in a learning round, with the remaining parameters fixed. Concretely, suppose \( \lambda_i \) is the parameter being optimized. Since \( \pi \) is not differentiable, we obtain the optimal value of \( \lambda_i \) by grid search: we evaluate \( \lambda_i \) with its values ranging over \( \{0, 1/L, 2/L, \ldots, 1\} \), where \( L \) is an integer parameter controlling the quantization granularity. The value which maximizes \( \pi \) is selected as the new value of \( \lambda_i \). Then, \( \lambda_{i+1} \) is activated and the same procedure is repeated. The loop continues until \( \pi \) no longer increases. To ensure the convexity constraint, we re-normalize the parameters after each iteration.

Given the large array of tags in the social web, where well-labeled training data are often unavailable, it is worthwhile to consider unsupervised fusion algorithms. Given no prior information concerning \( \{\lambda_i\} \), we should choose uniform weights, meaning we make the least assumptions about things we do not know [50]. Following this thought, we consider the following two simple fusion algorithms: Average and Borda Count.

**Unsupervised Fusion I: Average.** For early fusion, the visual similarity is the average of \( \{\text{sim}_i(x, x')\} \). To cope with the potential scale issue, we re-scale the
similarity scores of individual features by min-max normalization as used by Makadia et al. [80]. For late fusion, the corresponding $G^i_L(x, w)$ is simply the average of $\{g_i(x, w)\}$.

**Unsupervised Fusion II: Borda.** The Borda Count algorithm is well recognized as a solid choice for combining rankings generated by multiple sources of evidence [?,89]. The only difference between Borda and Average is that Borda quantizes (continuous) scores (which are visual similarities in early fusion or tag relevance values in late fusion) into discrete ranks. It is thus more robust to outliers when compared to Average. To apply the Borda algorithm, for early fusion we replace $\text{sim}_i(x, x')$ in (3.2) by the corresponding rank-based scores. For late fusion, $g_i(x, w)$ in (3.4) is replaced.

The unsupervised fusion methods do not need a training step. To train a supervised fusion model, a learning step is required. For late fusion, when $\{g_i(x, w)\}$ have been pre-computed, evaluating one of the $m$ parameters in $\Lambda$ in an iteration involves computing $G^i_L(x, w)$ for each image in $D_w$, with a time complexity of $O(|D_w| \cdot m)$. Subsequently, sorting $D_w$ requires $O(|D_w| \cdot \log |D_w|)$. As there are $L$ values to evaluate in an iteration, the complexity per iteration is $O(L \cdot |D_w| \cdot (m + \log |D_w|))$. The overall complexity for optimizing the late fusion model is $O(T \cdot L \cdot |D_w| \cdot (m + \log |D_w|))$, where $T$ is the number of iterations. For early fusion, suppose that the neighbor sets $\{S_{x,f,i,k}\}$ have been pre-computed. Evaluating a specific parameter in an iteration involves computing (3.3) for each image in $D_w$. The most computationally intensive part of (3.3) is obtaining $S_{x,A,k}$. Computing the fused similarity (3.2) has an order of $O(|S| \cdot m)$, and selecting the $k$ neighbors from $S$ by partial sort needs $O(|S| \cdot \log k)$. So getting $S_{x,A,k}$ for each image in $D_w$ has a time complexity of $O(|S| \cdot (m + \log k))$. Evaluating a given parameter has an order of $O(|D_w| \cdot |S| \cdot (m + \log k))$ plus $O(|D_w| \cdot \log |D_w|)$ for sorting $D_w$. Note that $\log |D_w| \ll |S|$ in general, meaning we can ignore the sorting cost. So the complexity per iteration is $O(L \cdot |D_w| \cdot |S| \cdot (m + \log k))$, and the overall complexity for optimizing the early fusion model is $O(T \cdot L \cdot |D_w| \cdot |S| \cdot (m + \log k))$. Hence, training a supervised late fusion model is far more efficient than its early fusion counterpart.

### 3.5 Experimental Setup

Given the large-scale nature of our problem, it is impractical to explicitly verify the assumptions on early and late fusion, as this would require ground truth on every image from the neighbor sets of all visual features. More importantly, our goal is social image search. Therefore, we evaluate the performance of the entire search systems.
3.5. Experimental Setup

3.5.1 Data Sets

Social Images for Tag Relevance Estimation $S$. We use a publicly available set of 3.5M social-tagged images†, collected from Flickr in a random manner in our previous work [64]. Since batch-tagged images tend to be visually redundant, we remove such images. Also, we remove images having no tags corresponding to WordNet. After this preprocessing step, we obtain a compact set of 80K images as an instantiation of $S$.

Ground-truth data. We choose NUS-WIDE contributed by Chua et al. [22]. This widely used ground-truth data consists of Flickr images with manually verified annotations for 81 diverse visual concepts. We again remove batch-tagged images, and preserve images whose social tags contain at least one of the 81 concepts. For social image search, supervised fusion models shall be trained using existing data, and applied to data which will be uploaded to the social web in the future. To simulate such a scenario, we divide images in NUS-WIDE into two subsets in terms of their DateUploaded property: NUS-past and NUS-future. We use NUS-past for training (64,048 images), and NUS-future for testing (64,049 images). The statistics of the data are given in Table 3.1.

3.5.2 Social Image Search Experiments

For each of the 81 concepts, we conduct tag-based image search. We sort images labeled with the concept in descending order by (fused) tag relevance values. The two fusion schemes, early and late, and the three fusion algorithms, Coord-Ascent, Average, and Borda, result in six fusion methods. We use EarlyFusion-Average to represent the combination of early fusion and the Average algorithm. In a similar manner, we name the other five methods EarlyFusion-Borda, EarlyFusion-Coord-Ascent, LateFusion-Average, LateFusion-Borda, and LateFusion-Coord-Ascent. For a more comprehensive comparison, we also report image search performance using three simple metadata features: DateUploaded, Views, and TagNum. Given an image, Views indicates how many times the image has been viewed. TagNum is the number of social tags assigned to the image. For DateUploaded and Views, we rank images in descending order to favor freshness and popularity. For TagNum, we sort images in ascending order to penalize over-tagged images.

Evaluation Criteria. We adopt average precision (AP), a common choice for evaluating visual search engines [105]. For an overall assessment, we report mean average precision (MAP), the average of AP scores of all concepts. Because we aim to answer whether tag relevance estimation can benefit from fusion, we consider the performance difference between single measurements of tag relevance and different fusion methods more informative to draw conclusions.

†Data available at http://staff.science.uva.nl/~xirong/tagrel/
Chapter 3. Tag Relevance Fusion for Social Image Search

Table 3.1: Statistics of the ground-truth data [22] used in our experiments. The large variance in the tagging accuracy of diverse concepts implies the importance of tag relevance estimation for social image search.

<table>
<thead>
<tr>
<th>Per concept</th>
<th>Training data</th>
<th>Test data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>min</td>
<td>max</td>
</tr>
<tr>
<td>No. images</td>
<td>125</td>
<td>7,749</td>
</tr>
<tr>
<td>No. relevant images</td>
<td>7</td>
<td>7,212</td>
</tr>
<tr>
<td>Tagging accuracy</td>
<td>0.06</td>
<td>0.93</td>
</tr>
</tbody>
</table>

The number of images for a given concept is the number of images labeled with the concept by social tagging. The number of relevant images is the number of images labeled with and relevant to the concept. (Social) Tagging accuracy is the number of relevant images divided by the number of images.

3.5.3 Implementation

Base tag relevance estimators \{g_i(x, w)\}. The base estimators are specified by the visual features \(f\) and the number of neighbors \(k\). As we have discussed in Section 3.4, \(k\) does not contribute significantly for diversifying the base estimators. So we empirically fix \(k\) to be 500. Concerning \(f\), we choose the following four visual features which describe image content from different perspectives: COLOR, CSLBP, GIST, and DSIFT. COLOR is a 64-d global feature, combining the 44-d color correlogram [47], the 14-d texture moments [145], and the 6-d RGB color moments. CSLBP is a 80-d center-symmetric local binary pattern histogram [44], capturing local texture distributions. GIST is a 960-d feature describing dominant spatial structures of a scene by a set of perceptual measures such as naturalness, openness, and roughness [88]. DSIFT is a 1024-d bag-of-keypoints histogram depicting local information of the visual content. We adopt dense sampling for keypoint localization and the SIFT descriptor for keypoint description [122]. As aforementioned, for all features we use the Euclidian distance to measure visual similarity. We denote the four tag relevance estimators by the corresponding feature names.

Parameters for Supervised Fusion. As we discussed in Section 3.4.2, optimizing the early fusion model is computationally much more expensive than optimizing the late fusion model. To keep them comparable, for each concept we randomly sample at maximum 500 training examples to form the training data \(D_w\). We empirically set \(L = 100\). On a dual-quad-core compute node with 2.4 GHz CPUs and 24 GB memory, training \(G^L_\Lambda(x, w)\) takes about 49 minutes per iteration. Training \(G^L_\Lambda(x, w)\) is far more efficient: approximately 0.3 seconds per iteration. Concerning factors affecting the trained models, we observe that the choice of \(L\) and the amount of training data are less dominant compared to the divergence between training and test data.
3.6 Results

Single Tag Relevance versus Metadata Features. To better show the performance difference between different methods, we use the average performance of the four base tag relevance estimators as a reference point, which has an MAP of 0.637. As shown in Fig. 3.2, we find among the three metadata features that TagNum with an MAP of 0.546 is the best, followed by Views and DateUploaded. They are all inferior to the base estimators. This result confirms that image search using learned tag relevance is superior to image search using original tags [64]. Concerning the base estimators, as they use four distinct features, their performance varies with a standard deviation of 0.013. Notice that the deviation is approximately 2.1% of the average performance. Thus, we conclude that the neighbor voting algorithm is robust with respect to the visual features used.

Tag Relevance Fusion versus Single Tag Relevance. As shown in Fig. 3.2, tag relevance fusion leads to better image search performance. For instance, EarlyFusion-Borda and LateFusion-Average obtain an absolute improvement of 0.045 and 0.052, respectively, which is 7.1% and 8.2% better than the average performance of the single tag relevance methods. For a comprehensive understanding, we make a per-concept comparison, as illustrated in Fig. 3.3. For 78 out of the 81 concepts, the search performance is improved by both EarlyFusion-Borda and LateFusion-Average.

Further, for each concept we check the best performer among the four base esti-
Figure 3.3: Tag Relevance Fusion versus Single Tag Relevance: A per-concept comparison. The concepts are sorted in descending order by the absolute improvement of the LateFusion-Coord-Ascent method. Best viewed in color.

mators. We find that for 22 concepts COLOR is the best, for 10 concepts CSLBP, for 31 concepts GIST, and for 18 concepts DSIFT. For every concept, we compare
EarlyFusion-Coord-Ascent and LateFusion-Coord-Ascent with the concept’s best performer. For 55 out of the 81 concepts, EarlyFusion-Coord-Ascent outperforms the best performers, which vary per concept. For 65 concepts, LateFusion-Coord-Ascent is better than the best performers. These results show the effectiveness of tag relevance fusion for social image search.

Comparing Unsupervised Fusion Methods. The overall results of the six fusion methods are given in Table 3.2. Concerning the two unsupervised fusion algorithms, we observe that Borda is better than Average for early fusion, while Average is more suited for late fusion. As shown in Fig. 3.4, for 70 concepts EarlyFusion-Borda is better than EarlyFusion-Average. The result shows that the rank-based normalization is more effective than the min-max normalization for unsupervised fusion of visual similarities. In contrast, LateFusion-Average outperforms LateFusion-Borda for 68 concepts. This result is mainly due to the fact that the base estimators already include an effect of smoothing by quantizing the visual neighborhood via neighbor voting. Further quantization by the Borda algorithm makes tag relevance estimates less discriminative. Only when some base estimators yield large yet inaccurate values such as COLOR for “rainbow” as illustrated in Fig. 3.6(b), Borda is preferred. We also observe the limitations of late fusion for addressing concepts which are rarely tagged. Consider “earthquake” for instance. There are only 113 images labeled with the concept in our social collection $S$. The base estimators mostly yield zero scores for the concept. Late Fusion does not add much in this case. In contrast, by directly manipulating the neighbor sets, EarlyFusion-average yields the best result for “earthquake”. In general, we consider LateFusion-Average the best choice for unsupervised fusion, for its competitive performance and its flexibility in adding new base estimators.

Comparing Supervised Fusion Methods. As shown in Table 3.2, the supervised methods achieve the best performance for both early and late fusion. In theory, given enough training data which well represents (unseen) test data, EarlyFusion-Coord-Ascent should be better than LateFusion-Coord-Ascent, as the latter implies quantization at every decision point and thus a possible loss of information. However, given the dynamic nature of social data, training data from the past is sometimes inadequate to represent future data. In such a cross-data scenario, late fusion with

<table>
<thead>
<tr>
<th>Fusion Algorithms</th>
<th>Early</th>
<th>Late</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coord-Ascent</td>
<td>0.683</td>
<td>0.695</td>
</tr>
<tr>
<td>Average</td>
<td>0.670</td>
<td>0.689</td>
</tr>
<tr>
<td>Borda</td>
<td>0.682</td>
<td>0.673</td>
</tr>
</tbody>
</table>
Chapter 3. Tag Relevance Fusion for Social Image Search

Figure 3.4: Comparing the two unsupervised fusion algorithms: Average versus Borda. Borda corresponds to the reference line $x=0$. Falling at the left side of the line means Borda is better, while falling at the right side means Average is better. For a better view of the data, we sort concepts for early and late fusion separately. While Borda is better than Average for early fusion, Average is more suited for late fusion.

Supervised versus Unsupervised Fusion Methods. Comparing the supervised methods to their unsupervised alternatives, for some concepts such as “surf” and “rainbow” where there is large variance in the performance of the base estimators, the relative improvement is up to 37.3% and 27.4%. However, when averaging over all concepts, the improvement is marginal. The EarlyFusion-Coord-Ascent method beats EarlyFusion-Average with a relative gain of 1.9%, and LateFusion-Coord-Ascent improves LateFusion-Average with a relative gain of 0.9% only. This result seems counter-intuitive, as one would expect a larger improvement from supervised learning. To reveal whether the result is caused by a few outlier concepts, we also look into individual concepts. As shown in Fig. 3.5, although for 41 out of the 81 concepts LateFusion-Coord-Ascent improves over LateFusion-Average, there are only 6 concepts which have a relative improvement of more than 5%.

quantization tends to be more robust than early fusion. Indeed, our experiment indicates that early and late fusion perform similarly, with even a small increase for the latter.
3.6. Results

Figure 3.5: Comparing supervised and unsupervised fusion algorithms: Coord-Ascent versus Average. Average corresponds to the reference line $x=0$. Falling at the left side of the line means Average is better, while falling at the right side means Coord-Ascent is better. For a better view of the data, we sort concepts for early and late fusion separately. For both early and late fusion, the coordinate ascent learning algorithm obtains parameters better than the uniform weights. Because late fusion is more robust than early fusion, the improvement in late fusion is smaller than the improvement in early fusion.

We attribute such counter-intuitive results to the following two reasons. First, due to vagaries of social data, the models optimized on the past data tend to be suboptimal for the future data. Consider the concept “soccer” for instance. The CSLBP and DSIFT estimators have AP scores of 0.263 and 0.389 on the training data respectively, while their performance on the test data is much lower with AP scores of 0.152 and 0.130. In contrast, the performance of the other two base estimators is relatively stable when crossing datasets. As a consequence, the optimized model indeed makes the performance degenerate when compared to LateFusion-Average. Second, different from traditional learning-to-fuse scenarios where features or rankers might be just better than random guess [35,117], the features employed in this study were intellectually designed and shown to be effective. As shown in Fig. 3.2, the base estimators already provide a strong starting point. Moreover, distinct features result in complementary neighbor sets for early fusion and complementary tag relevance
Chapter 3. Tag Relevance Fusion for Social Image Search

Figure 3.6: Image search results of the concept “rainbow”. The top 15 results of each method are shown. A red border indicates a false positive result. Best viewed in color.

estimates for late fusion. Therefore, though unsupervised fusion is suboptimal for some concepts, it is practically as effective as supervised fusion in general.

3.7 Discussions and Conclusions

Estimating the relevance of social tags with respect to the visual content is essential for social image search but challenging. In this chapter we introduce tag relevance fusion as an extension to methods for tag relevance estimation. Using the neighbor voting algorithm to instantiate base tag relevance estimators [64], we have conducted a systematic study on early and late tag relevance fusion schemes. Image search experiments on a large benchmark show that compared to a single measurement of tag relevance, fusing multiple tag relevance driven by diverse features improves the image search accuracy. By a coordinate ascent based supervised fusion, we obtain
a relative improvement of 8.2% in terms of mean average precision. As revealed by overall and concept-by-concept analysis, tag relevance fusion robustifies tag relevance estimation for social image search.

The two fusion schemes each have their merit. Early fusion which directly manipulates the neighbor sets is more effective for addressing concepts rarely tagged. Late fusion is more robust to differences between the data on which the method is trained and the data to which it is applied. As the latter is what is mostly encountered when exploiting social data, it is the method of choice.

An exciting observation is that the simple unsupervised fusion algorithms perform comparably to their supervised counterparts. LateFusion-Average, which simply averages multiple tag relevance estimates, is effective, with a loss of 1.9% only, when compared to the best supervised fusion method. Moreover, it is more flexible than early fusion methods for incorporating novel tag relevance estimators. We recommend LateFusion-Average as a practical mechanism to exploit diverse tag relevance estimates for social image search.