Learning the Consensus on Visual Quality for Next-Generation Image Management

Ritendra Datta, Jia Li, and James Z. Wang
The Pennsylvania State University, University Park, PA 16802, USA
{datta, jiali, jwang}@psu.edu

ABSTRACT

While personal and community-based image collections grow by the day, the demand for novel photo management capabilities grows with it. Recent research has shown that it is possible to learn the consensus on visual quality measures such as *aesthetics* with a moderate degree of success. Here, we seek to push this performance to more realistic levels and use it to (a) help select high-quality pictures from collections, and (b) eliminate low-quality ones, introducing appropriate performance metrics in each case. To achieve this, we propose a sequential arrangement of a weighted linear least squares regressor and a naive Bayes' classifier, applied to a set of visual features previously found useful for quality prediction. Experiments on real-world data for these tasks show promising performance, with significant improvements over a previously proposed SVM-based method.

Categories and Subject Descriptors

H.4.m [**Information Systems Applications**]: Miscellaneous

General Terms

Algorithms, Experimentation, Performance.

1. INTRODUCTION

The immense popularity of photo-sharing communities (e.g., Flickr, Photobucket, Photo.net) and social-networking platforms (e.g., Facebook, Myspace) has made it imperative to introduce novel media management capabilities, which in turn may help to stay competitive in these crowded markets. In the case of visual media management, areas such as content-based image classification and retrieval [7], automatic annotation [1, 5], and image watermarking [2] for rights management have been extensively studied. Complementing some of these techniques, our goal is to be able to automatically assess high-level visual quality (unlike low-level quality such as noise/quantization level), so as to facilitate *quality-based image management*. Among other things,

MM'07, September 23–28, 2007, Augsburg, Bavaria, Germany.

Figure 1: Example images from Photo.net where the consensus aesthetics score ≥ 6 (above), and ≤ 4 (below), on $1-7$.

it can help (a) *select* high-quality images from a collection for browsing, for front-page display, or as representatives, (b) *enhance image search* by pushing images of higher quality up the ranks, and (c) *eliminate* low-quality images under space constraints (limited Web space, mobile device, etc.) or otherwise. Visual quality here can be based on criteria such as *aesthetics* (Photo.net, see Fig. 1) or *interestingness* (Flickr), and these can be either *personalized* (individuals treated separately), or *consensus-based* (scores averaged over population). A major deterrent to research in this direction has been the difficulty to precisely define their characteristics, and to relate them to low-level visual features. One way around this is to ignore philosophical/psychological aspects, and instead treat the problem as one of data-driven statistical inferencing, similar to user preference modeling in recommender systems [6].

Recent work [3] on aesthetics modeling for images has, however, given hope that it may be possible to empirically learn to distinguish between images of low and high aesthetic value[1]. A key result presented in that work is as follows. Using carefully chosen visual features followed by feature selection, a support vector machine (SVM) can distinguish between images rated > 5.8 and < 4.2 (on a 1-7 scale) with 70% accuracy and those rated ≥ 5.0 and < 5.0 with 64% accuracy, images being rated publicly by Photo.net users. There are two key concerns in the context of applicability of these results. (1) A 64% accuracy in being able to distinguish (≥ 5.0,< 5.0) is not a strong-enough for real-world deployment in selecting high-quality pictures (if ≥ 5.0 implies high-quality, that is). (2) It is unclear how a 70% accuracy on a (> 5.8, < 4.2) question can be used to help photo management in any way. To address them, we make the following contributions in this paper: (A) Given a set

[1]Since the use of the word *aesthetics* in this context is subject to controversy, we simply treat it as one possible measure of visual quality.

of visual features known to be useful for visual quality, we propose a new approach to exploiting them for significantly improved accuracy in inferring quality. (B) We introduce a weighted learning procedure to account for the trust we have in each consensus score, in the training data, and empirically show consistent performance improvement with it. (C) We propose two new problems of interest that have direct applicability to image management in real-world settings. Our approach produces promising solutions to these problems.

2. PROPOSED APPROACH

Let us suppose that there are D visual features known (or hypothesized) to have correlation with visual quality (e.g., aesthetics, interestingness). An image I_k can thus be described by a feature vector $\vec{X}_k \in \mathbb{R}^D$, where we use the notation $X_k(d)$ to refer to component d of feature vector $\vec{X}_k$. For clarity of understanding, let us assume that there exists a *true* measure q_k of consensus on the visual quality that is intrinsic to each I_k. Technically, we can think of this true consensus as the asymptotic average over the entire population, i.e., $q_k = \lim_{Q\to\infty} \frac{1}{Q}\sum_{i=1}^{Q} q_{k,i}$, where $q_{k,i}$ is the i^{th} rating received. This essentially formalizes the notion of 'aesthetics in general' presented in [3]. This measurement is expected to be useful to the average user, while for those 'outliers' whose tastes differ considerably from the average, a personalized score is more useful - a case that best motivates recommender systems with individual user models.

In reality, it is impractical to compute this true consensus score because it requires feedback over the entire population. Instead, items are typically scored by a small subset of the population, and what we get from averaging over this subset is an estimator for q_k. If $\{s_{k,1}, \cdots, s_{k,n_k}\}$ is a set of scores provided by n_k unique users for I_k, then $\hat{q}_k = \frac{1}{n_k}\sum_{i=1}^{n_k} s_{k,i}$, where $\hat{q}_k$ is an estimator of q_k. In theory, as $n_k \to \infty$, $\hat{q}_k \to q_k$. Given a set of N training instances $\{(\vec{X}_1, \hat{q}_1), \cdots, (\vec{X}_N, \hat{q}_N)\}$, our goal is to learn a model that can help predict quality from the content of unseen images.

2.1 Weighted Least Squares Regression

Regression is a direct attempt at learning to emulate human ratings of visual quality, which we use here owing to the fact that it is reported in [3] to have found some success. Here, we follow the past work by learning a least squares linear regressor on the predictor variables $X_k(1), \cdots, X_k(D)$, where the dependent variable is the consensus score $\hat{q}_k$. We introduce *weights* to the regression process on account of the fact that $\hat{q}_k$ are only estimates of the true consensus q_k, with less precise estimates being less trustable for learning tasks. From classical statistics, we know that the *standard error of mean*, given by $\frac{\sigma}{\sqrt{n}}$, decreases with increasing sample size n. Since $\hat{q}_k$ is a mean estimator, we compute the weights w_k as a simple increasing function of sample size n_k,

$$w_k = \frac{n_k}{n_k + 1},\ k = 1, \cdots, N \tag{1}$$

where $\lim_{n_k\to\infty} w_k = 1$, $w_k \in [\frac{1}{2}, 1)$. The corresponding parameter estimate for squared loss is written as

$$\vec{\beta}^* = \arg\min_{\vec{\beta}} \frac{1}{N}\sum_{k=1}^{N} w_k \left(\hat{q}_k - \left(\beta(0) + \sum_{d=1}^{D}\beta(d)X_k(d)\right)\right)^2$$

Given a $\vec{\beta}^*$ estimated from training data, the predicted score for an unseen image I having feature vector X is given by

$$q^{pred} = \beta^*(0) + \sum_{d=1}^{D}\beta(d)X(d) \tag{2}$$

Because weighted regression is relatively less popular than its unweighted counterpart, we briefly state an elegant and efficient linear algebraic [4] estimation procedure, for the sake of completeness. Let us construct an $N\times(D+1)$ matrix $\mathbf{X} = [\ \vec{\mathbf{1}}\ \mathbf{Z}^T\ ]$ where $\vec{\mathbf{1}}$ is a N-component vector of ones, and $\mathbf{Z} = [\vec{X}_1 \cdots \vec{X}_N]$. Let $\vec{q}$ be a $N\times 1$ column matrix (or vector) of the form $(\hat{q}_1 \cdots \hat{q}_N)^T$, and $\mathbf{W}$ is an $N \times N$ diagonal matrix consisting of the weights, i.e., $W = diag\{w_1, \cdots, w_N\}$. In the unweighted case of linear regression, the parameter estimate procedure is given by $\vec{\beta}^* = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\vec{q} = \mathbf{X}^\dagger\vec{q}$, where $\mathbf{X}^\dagger$ is the *pseudoinverse* in the case of linearly independent columns. The weighted linear least squares regression parameter set, on the other hand, is estimated as below:

$$\vec{\beta}^* = (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\vec{q} \tag{3}$$

Letting $\mathbf{V} = diag\{\sqrt{w_1}, \cdots, \sqrt{w_N}\}$, such that $\mathbf{W} = \mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T$, we can re-write Eq. 3 in terms of pseudoinverse:

$$\begin{aligned}\vec{\beta}^* &= (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\vec{q} \\ &= ((\mathbf{V}\mathbf{X})^T(\mathbf{V}\mathbf{X}))^{-1}(\mathbf{V}\mathbf{X})^T\mathbf{V}\vec{q} \\ &= (\mathbf{V}\mathbf{X})^\dagger\mathbf{V}\vec{q}\end{aligned} \tag{4}$$

This form may lead to cost benefits. Note that the weighted learning process does not alter the inference step of Eq. 2.

2.2 Naive Bayes' Classification

The motivation for having a naive Bayes' classifier was to be able to complement the linear model with a probabilistic one, based on the hypothesis that they have non-overlapping performance advantages. The particular way of fusing regression and classification will become clearer shortly. For this, we assume that by some predetermined threshold, the (consensus) visual quality scores $\hat{q}_k$ can be mapped to binary variables $\hat{h}_k \in \{-1, +1\}$. For simplification, we make a conditional independence assumption on each feature given the class, to get the following form of the naive Bayes' classifier:

$$Pr(H \mid X(1), \cdots, X(D)) \propto Pr(H)\prod_{d=1}^{D} Pr(X(d) \mid H) \tag{5}$$

The inference for an image I_k with features $\vec{X}_k$ involves a simple comparison of the form

$$\hat{h}_k = \arg\max_{h\in\{-1,+1\}} Pr(H = h)\prod_{d=1}^{D} Pr\Big(X_k(d) \mid H = h\Big) \tag{6}$$

The training process involves estimating $Pr(H)$ and $Pr(X(d)|H)$ for each d. The former is estimated as follows:

$$\Pr(H = h) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{I}(\hat{h}_i = h) \tag{7}$$

where $\mathcal{I}(\cdot)$ is the indicator function. For the latter, parametric distributions are estimated for each feature d given class. While Gaussian mixture models seem appropriate for complicated feature values (e.g., too high or too low brightness are both not preferred), here we model each of them using single component Gaussian distributions, i.e.,

$$X(d) \mid (H = h)\ \sim\ \mathcal{N}(\mu_{d,h}, \sigma_{d,h}),\ \forall d, h, \tag{8}$$

where the Gaussian parameters $\mu_{d,h}$ and $\sigma_{d,h}$ are the mean and std. dev. of the feature value X_d over those training samples k that have $\hat{h}_k = h$. Performing weighted parameter estimation is possible here too, although in our experiments we restricted weighting learning to regression only.

2.3 Selecting High-quality Pictures

Equipped with the above two methods, we are now ready to describe our approach to selecting high-quality images. First we need a definition for 'high-quality'. An image I_k is considered to be visually of high-quality if its estimated consensus score, as determined by a subset of the population, exceeds a predetermined threshold, i.e., $\hat{q}_k \geq HIGH$. Now, the task is to automatically select T high-quality images out of a collection of N images. Clearly, this problem is no longer one of classification, but that of retrieval. The goal is to have high *precision* in retrieving pictures, such that a large percentage of the T pictures selected are of high-quality. To achieve this, we perform the following:

1. A weighted regression model (Sec. 2.1) is learned on the training data.
2. A naive Bayes' classifier (Sec. 2.2) is learned on training data, where the class labels $\hat{h}_k$ are defined as
$$\hat{h}_k = \begin{cases} +1 & \text{if } \hat{q}_k \geq HIGH \\ -1 & \text{if } \hat{q}_k < HIGH \end{cases}$$
3. Given an unseen set of N test images, we get predict consensus scores $\{\hat{q}_1, \cdots, \hat{q}_N\}$ using the weighted regression model, which we sort in *descending* order.
4. Using the naive Bayes' classifier, we start from the top of the ranklist, selecting images for which the predicted class is $+1$, i.e., $\hat{h} = +1$, *and* $\frac{Pr(H=+1|X(1),\cdots,X(D))}{Pr(H=-1|X(1),\cdots,X(D))} > \theta$, until T of them have been selected. This filter applied to the ranked list therefore requires that only those images at the top of the ranked list that are also classified as high-quality by the naive Bayes' (and convincingly so) are allowed to pass. For our experiments, we chose $\theta = 5$ arbitrarily and got satisfactory results.

2.4 Eliminating Low-quality Pictures

Here, we first need to define 'low-quality'. An image I_k is considered to be visually of low-quality if its consensus score is below a threshold, i.e., $\hat{q}_k \leq LOW$. Again, the task is to automatically filter out T low-quality images out of a collection of N images, as part of a space-saving strategy (e.g., presented to the user for deletion). The goal is to have high precision in eliminating low-quality pictures, with the added requirement that as few high-quality ones (defined by threshold $HIGH$) be eliminated in the process as possible. Thus, we wish to eliminate as many images having score $\leq LOW$ as possible, while keeping those with score $\geq HIGH$ low in count. Here, steps 1 and 2 of the procedure are same as before, while steps 3 and 4 differ as follows:

1. In Step 3, instead of sorting the predicted consensus scores in descending order, we do so in *ascending* order.
2. In Step 4, we start from the top of the ranklist, selecting images for which the predicted class is -1 (not $+1$, as before), by a margin. This acts as as a two-fold filter: (a) low values for the regressed score ensure preference toward selecting low-quality pictures, and (b) a predicted class of -1 by the naive Bayes' classifier prevents those with $HIGH$ scores from being eliminated.

3. EXPERIMENTS

All experiments are performed on the same dataset obtained from Photo.net that was used in [3], consisting of 3581 images, each rated publicly by one or more Photo.net users on a $1-7$ scale, on two parameters, (a) aesthetics, and (b) originality. As before, we use the aesthetics score as a measure of quality. While individual scores are unavailable, we do have the average scores $\hat{q}_k$ for each image I_k, and the no. of ratings n_k given to it. The score distribution in the $1-7$ range, along with the distribution of the per-image number of ratings, is presented in Fig. 2. Note that the lowest average score given to an image is 3.55, and that the number of ratings has a *heavy-tailed* distribution. The same 56 visual features extracted in [3] (which include measures for brightness, contrast, depth-of-field, saturation, shape convexity, region composition, etc.) are used here as well, but without any feature selection being performed. Furthermore, nonlinear powers of each of these features, namely their squares, cubes, and square-roots, are augmented with them to get $D = 224$ dimensional feature vectors describing each image.

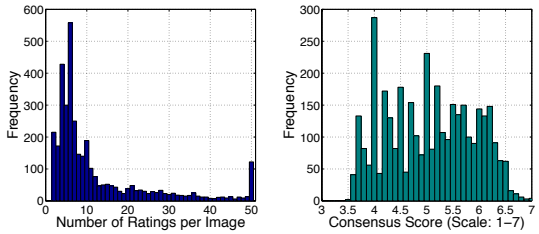

Figure 2: Distributions of no. of ratings (left) and scores (right) in the Photo.net dataset.

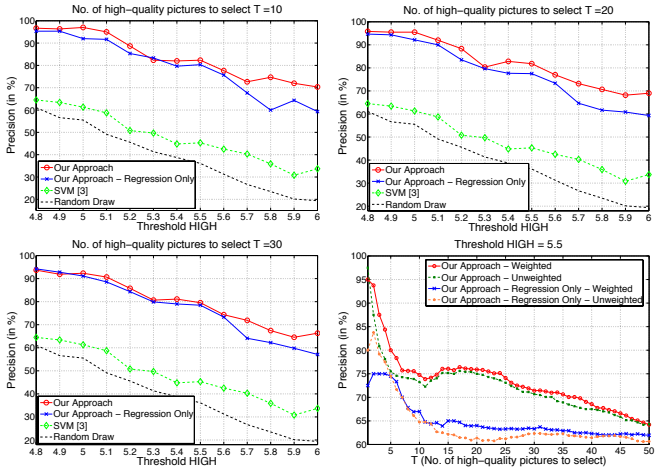

Figure 3: Precision in selecting high-quality images, shown here for three selection set sizes, $T = 10$, 20, and 30. *Bottom-right:* Impact of using weighted model estimation vs. their unweighted counterparts, with $HIGH$ fixed and T varying.

3.1 Selecting High-quality Pictures

Using the procedure described in Sec. 2.3, we perform experiments for selection of high-quality images for different values of $HIGH$, ranging over $4.8 - 6.0$ out of a possible 7, in intervals of 0.1. In each case, 1000 images are drawn uniformly at random from the 3581 images for testing, and the remaining are used for training the regressor and the classifier. The task here is to select $T = 5$, 10, and 20 images out of the pool of 1000 (other values of $T \leq 50$ showed similar trends), and measure the *precision* $= \frac{\#(\text{high-quality images selected})}{\#(\text{images selected})}$, where the denominator is a chosen T. We compare our approach with three baselines. First, we use only the regressor and not the subsequent classifier (named 'Regression only'). Next we use an SVM, as

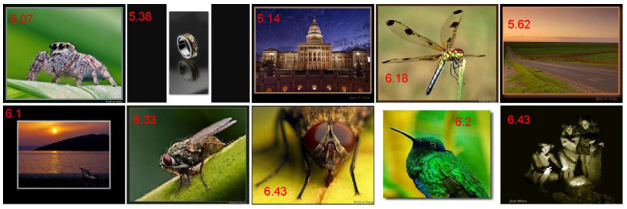

Figure 4: A sample instance of $T = 10$ images selected by our approach, for $HIGH = 5.5$. The actual consensus scores are shown in red, indicating an 80% precision in this case.

originally used in [3], to do a $(< HIGH, \geq HIGH)$ classification to get a fixed performance independent of T (named 'SVM'), i.e., the SVM simply classifies each test image, and therefore regardless of the number of images (T) to select, performance is always the same. Finally, as a worst-case bound on performance, we plot the precision achieved on picking any T images at random (named 'Random Draw'). This is also an indicator of the proportion of the 1000 test images that actually are of high-quality on an average. Each plot in Fig. 3 are averages over 50 random test sets.

We notice that our performance far exceeds that of the baselines, and that combining the regressor with the naive Bayes' in series pushes performance further, especially for larger values of $HIGH$ (since the naive Bayes' classifier tends to identify high-quality pictures more precisely). For example, when $HIGH$ is set to 5.5, and $T = 20$ images are selected, on an average 82% are of high-quality when our approach is employed, in contrast to less than 50% using SVMs. For lower thresholds, the accuracy exceeds 95%. In the fourth graph (bottom-right), we note the improvement achieved by performing weighted regression instead of giving every sample equal importance. Performed over a range of $HIGH$ values, these averaged results confirm our hypothesis about the role of 'confidence' in consensus modeling. For illustration, we present a sample instance of images selected by our approach for $T = 10$ and $HIGH = 5.5$, in Fig. 4, along with their ground-truth consensus scores.

3.2 Eliminating Low-quality Pictures

Here again, we apply the procedure presented in Sec. 2.4. The goal is to be able to eliminate T images such that a large fraction of them are of low-quality (defined by threshold LOW) while as few as possible images of high-quality (defined by threshold $HIGH$) get eliminated alongside. Experimental setup is same as the previous case, with 50 random test sets of 1000 images each. We experimented with various values of $T \leq 50$ with consistent performance. Here we present the cases of $T = 25$ and 50, fix $HIGH = 5.5$, while varying LOW from $3.8 - 5.0$. Along with the metric $precision = \frac{\#(\text{low-quality images eliminated})}{\#(\text{images eliminated})}$, also computed in this case is $error = \frac{\#(\text{high-quality images eliminated})}{\#(\text{images eliminated})}$. Measurements over both these metrics, with varying LOW threshold, and in comparison with the 'Regression Only', 'SVM', and 'Random Draw', are presented in Fig. 5.

These results are very encouraging, as before. For example, it can be seen that when the threshold for low-quality is set to 4.5, and 50 images are chosen for elimination, our approach ensures $\sim 65\%$ of them to be of low-quality, with only $\sim 9\%$ to be of high-quality. At higher threshold values, precision exceeds 75%, while error remains roughly the same. In contrast, the corresponding SVM figures are 43%

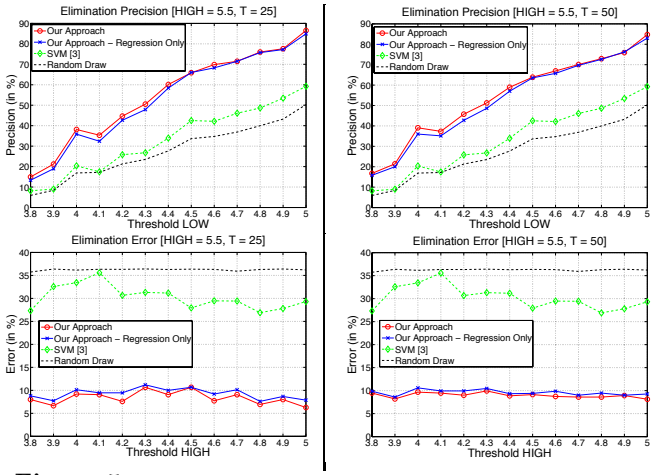

Figure 5: *Above:* Precision in eliminating low-quality images, shown here for two set sizes, namely $T = 25$ and 50. *Below:* The corresponding errors, made by eliminating high-quality images in the process.

and 28% respectively. We also note that the performance with using naive Bayes' in conjunction with regression does improve performance on both metrics, although not to the extent we see in high-quality picture selection. While not shown here, we found similar improvements as before with using the weighted methods over the unweighted ones. In general, our approach produces lesser guarantees in elimination of low-quality than selection of high-quality.

4. CONCLUSIONS

We have presented a simple approach to selecting high-quality images and eliminating low-quality ones from image collections, quality being defined by population consensus. Experiments show vast improvement over a previously proposed SVM-based approach. It is found that the same visual features proposed in [3] can show much more promising results when exploited by a different approach. Weighting the training data by confidence levels in the consensus scores is also found to consistently improve performance. The key to this success lies not necessarily in a better classifier, but in the fact that for these problems, it suffices to identify the extremes in visual quality, for a subset of the images, accurately.

5. REFERENCES

[1] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, 2002.

[2] I. Cox, J. Kilian, F. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Trans. Image Processing*, 6(12):1673–1687, 1997.

[3] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Studying aesthetics in photographic images using a computational approach. In *Proc. ECCV*, 2006.

[4] G. H. Golub and C. F. V. Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, Maryland, 1983.

[5] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(9):1075–1088, 2003.

[6] P. Resnick and H. Varian. Recommender systems. *Comm. of the ACM*, 40(3):56–58, 1997.

[7] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, , and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.