Visual Comparison of Images Using Multiple Kernel Learning for Ranking

Amr Sharaf[1]
amr.sharaf@alexu.edu.eg
Mohamed E. Hussein[*2]
mohamed.e.hussein@ejust.edu.eg
Mohamed A. Ismail[1]
maismail@alexu.edu.eg

[1] Alexandria University
El-Shatby, Alexandria 21526, Egypt

[2] Egypt-Japan University of Science and Technology (E-JUST)
New Borg Al-Arab, Alexandria 21934, Egypt

Abstract

Ranking is the central problem for many applications such as web search, recommendation systems, and visual comparison of images. In this paper, the multiple kernel learning framework is generalized for the learning to rank problem. This approach extends the existing learning to rank algorithms by considering multiple kernel learning and consequently improves their effectiveness. The proposed approach provides the convenience of fusing different features for describing the underlying data. As an application to our approach, the problem of visual image comparison is studied. Several visual features are used for describing the images and multiple kernel learning is adopted to find an optimal feature fusion. Experimental results on three challenging datasets show that our approach outperforms the state-of-the art and is significantly more efficient in runtime.

1 Introduction

Motivation Learning to rank is fundamental for many applications such as web search, recommendation systems, visual image comparison, and online advertisement. Without loss of generality, modeling visual image comparison is studied as an application for our ranking model.

Visual image comparison has been used extensively for a variety of applications [14, 24, 25, 26]. While traditional visual recognition approaches focus on object and activity recognition, recent work proposes models for visual image comparison based on their visual attributes. Visual Attributes are human-interpretable mid-level semantic concepts such as "furry", "natural", etc. that are shared across related categories. Relative attributes [24] were shown to be more effective than attribute scores from binary classifiers. For example, while it maybe difficult to determine the existence of a visual attribute in an image, it could be much easier to determine if one image exhibits a stronger visual attribute than the other (Top row of Figure 1).

Image comparison via relative attributes has emerged as a promising paradigm for many applications. In image search [14], relative attributes could be used by the user to describe

* Mohamed E. Hussein is currently with Egypt-Japan University of Science and Technology, on leave from his position at Alexandria University, Alexandria, Egypt.

Pages 95.1-95.13
DOI: https://dx.doi.org/10.5244/C.29.95

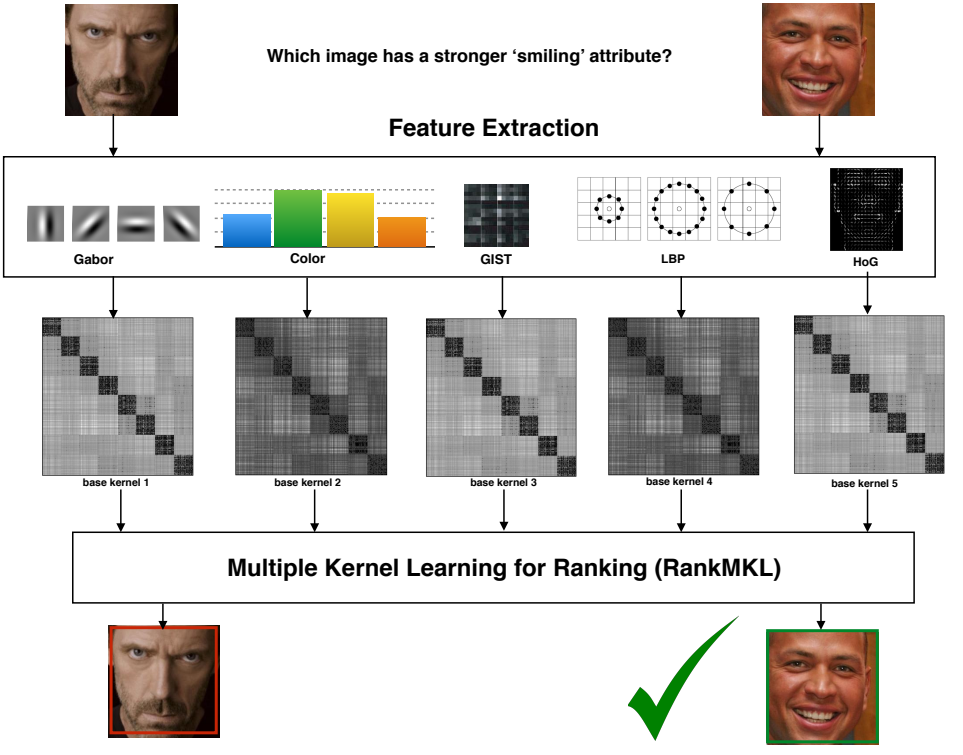


Figure 1: Illustration of our proposed approach. Given two images, we want to know which image has a stronger visual attribute than the other. Different features are extracted and multiple kernel learning is used for fusing the kernels from each feature set. RankMKL (3.3) is used for ranking the images.

which properties of exemplar images should be adjusted in order to improve the search results. In active learning [2, 25], relative attributes were used as a mode of communication for the human supervisor to provide an active learning machine with feedback when it predicts an incorrect label for an image. This allows a classifier to better learn from its mistakes, leading to accelerated discriminative learning of visual concepts even with few labeled images. In zero-shot learning [24], relative attributes were shown to yield significantly better zero-shot learning accuracy when compared to their binary counterparts.

Proposal We address the problem of modeling the visual comparison of images using relative attributes. We propose to solve the problem by fusing several visual features (histogram of oriented gradient [7], GIST descriptor [23], local binary patterns [22], filtering with a bank of Gabor filters [21], and color histograms), and use a kernel-based method for learning the model. While classical kernel-based methods are based on a single kernel, in practice it is often desirable to base the learned model on combinations of multiple kernels [1]. In this paper, the standard multiple kernel learning algorithm - used previously for object detection and image categorization - has been extended to learn an optimal feature fusion and build the ranking model.

Contributions We emphasize our main contributions over prior work:

1. The standard multiple kernel learning formulation is extended for the ranking problem and applied for modeling visual comparisons. To the best of our knowledge, this is the

first time that multiple kernel learning is used for ranking.

2. A novel approach is proposed for visual comparison of images based on feature fusion and multiple kernel learning. Experimental results on three datasets demonstrate the superiority of our proposed approach over the state-of-the-art.

2 Related Work

Learning comparison models and multiple kernel learning have been used extensively in the past several years. In this section, we briefly summarize the literature associated with each of these research areas.

Learning Comparison Models: Modeling visual comparisons could be categorized into two groups: linear methods, and non-linear methods. Parikh and Grauman [24] learned a linear ranking function for modeling relative attributes. Although the linear function performance may be inferior to nonlinear methods, it is useful to quickly produce a baseline model and its training time is faster. Li *et al*. [17] extended this approach for the non-linear case by using an ensemble of ranking trees to learn a model for each attribute. However, this approach fails to accommodate for fusing visual features of different types as it relies on learning a piecewise-linear ranking function which assumes equal importance for features of different types.

Information retrieval has also generated a vast volume in research literature for the task of learning to rank. Existing algorithms for learning to rank could be categorized into three groups according to the input representation: point-wise, pairwise, and list-wise algorithms. Point-wise algorithms [11, 27] predict the relevance of a point to a query by minimizing a regression loss. In pairwise ranking algorithms, the problem is approximated by a classification problem and learning a binary classifier that can tell which point is better in a given pair. The goal is to minimize average number of inversions in ranking. These approaches include rankSVM [13], RankBoost [9], RankNet [4], Deep Ranking [33], and GBRank [43]. List-wise algorithms such as Lambda-MART [34], and AdaRank [35] try to directly optimize a ranking performance measure over all queries in the training data. Our approach is based on an extension for the rankSVM [13] pairwise approach. One of the advantages of adopting the pairwise ranking approach is that most classification methods can be easily adapted to this formulation of the ranking problem. We take advantage of this property to extend the standard multiple kernel learning algorithm for the ranking problem.

Ranking models could also be categorized according to their locality. All the aforementioned methods could be regarded as global methods where a single ranking model is learned from the training data. Yu and Grauman [39] used a local learning model for the task of fine-grained image comparison. In this approach, a ranking model is learned on the fly from analogous training pairs and the learned model is used for ranking. Since the ranking model is learned from fine-grained neighbors, it was shown to be effective for fine-grained comparison of images; however, it has the disadvantage of being slow at test time since a new model has to be learned for every test pair.

In this paper, we present a non-linear, pairwise, and global approach based on rankSVM for modeling visual comparison of images. Our approach fuses multiple base kernels instead of one leading to a better discrimination between the compared images by combining multiple image features (*e.g*. shape, appearance, and texture features), and hence it outperforms the state-of-the-art both in coarse-grain and fine-grain visual comparisons.

Multiple Kernel Learning: we briefly review some of the related work for Multiple Kernel Learning in computer vision. We refer the reader to a more comprehensive study in [3, 10]. Multiple Kernel Learning (MKL) extends traditional kernel methods by combining multiple base kernels, leading to better representation and discrimination of samples. Since the introduction of MKL [1], it has been shown to improve several machine learning tasks such as classification [28, 31, 44], feature fusion [30, 41], variable selection [37], and dimensionality reduction [19, 20]. In computer vision, Vemulapalli *et al*. [32] used MKL for the classification of manifold features where the problem of learning a good kernel-classifier combination was formulated as a convex optimization problem and solved using a multiple kernel learning approach. Xu *et al*. [36] used MKL for complex event detection in videos by utilizing related exemplars. Chen *et al*. [5] used multiple kernel learning for the recognition of facial expressions in uncontrolled environments. To the best of our knowledge, this is the first time MKL is used for the image comparison task.

3 Proposed Approach

This section describes our proposed approach for modeling visual image comparisons using multiple kernel learning and feature fusion. The approach overview is described in section 3.1. Sections 3.2 and 3.3 show how the standard multiple kernel learning formulation is extended for the ranking problem.

3.1 Approach Overview

Given two images, we want to determine which image exhibits a particular visual attribute more than the other. Our approach works on a per attribute basis, thus a separate model is learned for each visual attribute. Figure 1 demonstrates the outline of our approach. The first step is to extract a set of features from each image. Several feature sets are selected to capture different visual cues in the image. To capture the image texture, we extract Local Binary Patterns (LBP) [22] and compute the response from a set of Gabor filters [21]. For capturing the shape and appearance of the images, GIST [23] and HoG [7] descriptors are used. Finally, we compute a color histogram in the LAB color space to capture the color information.

The second step is to fuse the different feature sets and learn the ranking model. For this task, a separate kernel function is computed for each set of features (i.e. we compute five different kernels). The computed kernels are considered as base kernels for our multiple kernel learning module. Using the multiple kernel learning algorithm described in section 3.3, we learn the optimal weights for creating a linear combination from the base kernels together with the optimal parameters for the ranking model.

A similar procedure takes place at test time, features are extracted, and the previously learned kernel weights and ranking model are used for evaluating the new test pair.

3.2 Support Vector Machines For Ranking

The training set in ranking SVM is given as $S = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ where $\mathbf{x}_i$ is a data point and y_i is an integer indicating the rank of $\mathbf{x}_i$, such that $\mathbf{x}_i \succ_s \mathbf{x}_j$ when $y_i < y_j$. We say $\mathbf{x}_i \succ_S \mathbf{x}_j$ if a vector $\mathbf{x}_i$ is ranked higher than $\mathbf{x}_j$. By defining the set of preference pairs as

$P = \{(i,j)|\mathbf{x}_i \succ_S \mathbf{x}_j\}$ with $p = |P|$, rankSVM [13] solves:

$$\begin{aligned} \underset{\mathbf{w},\varepsilon}{\text{minimize}} \quad & \{\frac{1}{2}\mathbf{w}^{\mathbf{T}}\mathbf{w} + C\sum_{(i,j)\in P}\varepsilon_{ij}\} \\ \text{subject to} \quad & \mathbf{w}.\phi(\mathbf{x}_i) \geq \mathbf{w}.\phi(\mathbf{x}_j) + 1 - \varepsilon_{ij}, \\ & \varepsilon_{ij} \geq 0, \forall (i,j) \in P, \end{aligned} \tag{1}$$

where $\mathbf{w}$ is the vector of parameters defining the optimal decision hyperplane and the regularization parameter $C > 0$ controls the generalization capabilities of the ranking function. ϕ is a function that maps data to a higher dimensional space. The loss term ε_{ij} in 1 is called L1 loss. If it is replaced by ε_{ij}^2 , we have L2 loss.

Several approaches have been proposed for solving 1 using kernel techniques [11, 16, 40]. In our approach, a mapping between the rankSVM and the traditional SVM problem is used as proposed by [16]. SVM [6] solves the following optimization problem:

$$\begin{aligned} \underset{\mathbf{w},\varepsilon}{\text{minimize}} \quad & \{\frac{1}{2}\mathbf{w}^{\mathbf{T}}\mathbf{w} + C\sum_{i=1}^{n}\varepsilon_i\} \\ \text{subject to} \quad & y_i\mathbf{w}^T\phi(\mathbf{x}_i) \geq 1 - \varepsilon_i, \\ & \varepsilon_i \geq 0, i = 1,...,n. \end{aligned} \tag{2}$$

The rankSVM optimization problem in 1 could be mapped to the SVM problem in 2 by defining:

$$\forall (i,j) \in P, y_{i,j} = 1, \phi_{i,j} = \phi(\mathbf{x}_i) - \phi(\mathbf{x}_j), \tag{3}$$

The problem 2 is solved by maximizing its dual on which the kernel trick can be applied:

$$\begin{aligned} \underset{\alpha}{\text{maximize}} \quad & \{\mathbf{1}^{\mathbf{T}}\alpha - \frac{1}{2}\alpha^T Q\alpha\} \\ \text{subject to} \quad & 0 \leq \alpha_{i,j} \leq C, \forall (i,j) \in P, \end{aligned} \tag{4}$$

where $\alpha \in \mathbb{R}^p$ are Lagrange multipliers indexed by pairs in P, $\mathbf{1} \in \mathbb{R}^p$ is a vector of ones, and $Q_{(i,j),(u,v)} = \phi_{i,j}^T\phi_{u,v}$ is a $p \times p$ symmetric matrix. The matrix is computed using the kernel function where:

$$Q_{(i,j),(u,v)} = k(\mathbf{x}_i,\mathbf{x}_u) + k(\mathbf{x}_j,\mathbf{x}_v) - k(\mathbf{x}_i,\mathbf{x}_v) - k(\mathbf{x}_j,\mathbf{x}_u) \tag{5}$$

and $k(\mathbf{x}_i,\mathbf{x}_j) = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$ is the kernel function representing the inner products after mapping the data to a higher dimension space.

Directly computing the matrix Q requires $O(n^4)$ kernel evaluations, Q can be factorized as $Q = AKA^T$, where K is the kernel matrix with $K = [k(\mathbf{x}_i,\mathbf{x}_j)]_{i,j=1,\cdots,n}$ and $A \in \mathbb{R}^{p\times n}$ is defined as follows:

$$A = \begin{bmatrix} & \cdots & i & \cdots & \cdots & j & \cdots \\ \vdots & & & & & & \\ (i,j) & 0 & 1 & 0 & 0 & -1 & 0 \\ \vdots & & & & & & \end{bmatrix} \tag{6}$$

That is, if $(i,j) \in P$ then the i^{th} entry of the corresponding row in A is 1, the j^{th} entry is -1, and all the other entries are zeros. Hence, computing Q requires $O(n^2)$ kernel evaluations. Storing Q is not needed any more and only the sparse matrix A and the kernel matrix K are kept in memory.

3.3 Multiple Kernel Learning for Ranking (RankMKL)

In the previous section, it was shown how to map the rankSVM optimization function in 1 to the SVM optimization problem 2. After mapping the problem to an SVM optimization, any of the multiple kernel learning algorithms proposed for the SVM classifier could directly be adopted. The Generalize Multiple Kernel Learning (GMKL) algorithm from [31] is used in our proposed approach.

Instead of using a single kernel matrix (K) for learning the ranking model, an optimal combination from several base kernels is learned, and the combination of the base kernels matrix $(K_{\mathbf{d}})$ is used for training the ranking model, where $k_{\mathbf{d}}(\mathbf{x}_i,\mathbf{x}_j) = \phi(\mathbf{x}_i)^T_{\mathbf{d}}\phi(\mathbf{x}_j)_{\mathbf{d}}$ represents the dot product in feature space ϕ and is parametrized by $\mathbf{d}$ such that:

$$k_{\mathbf{d}}(\mathbf{x}_i,\mathbf{x}_j) = f_{\mathbf{d}}(\{k_i(\mathbf{x}_i,\mathbf{x}_j)\}_{i=1}^{t}), \tag{7}$$

where t is the number of base kernels, $\mathbf{d} \in \mathbb{R}^t$ is the optimal kernel weights to be learned, and the combination function $f_{\mathbf{d}}$ can be a linear or a nonlinear function for combining the base kernels . Our goal is to learn the optimal values for $(\mathbf{d})$ together with the optimal values for the Lagrange multipliers (α) form 4 representing the learned ranking model. Accordingly, the objective function 4 is updated as follows:

$$\begin{aligned} \underset{\alpha}{\text{maximize}} \quad & \{\mathbf{1^T}\alpha - \frac{1}{2}\alpha^T Q_{\mathbf{d}}\alpha + r(\mathbf{d})\} \\ \text{subject to} \quad & 0 \le \alpha_{i,j} \le C, \forall (i,j) \in P, \\ & \mathbf{d} \ge 0, \end{aligned} \tag{8}$$

$$Q_{\mathbf{d},(i,j),(u,v)} = k_{\mathbf{d}}(\mathbf{x}_i,\mathbf{x}_u) + k_{\mathbf{d}}(\mathbf{x}_j,\mathbf{x}_v) - k_{\mathbf{d}}(\mathbf{x}_i,\mathbf{x}_v) - k_{\mathbf{d}}(\mathbf{x}_j,\mathbf{x}_u), \tag{9}$$

where both the regularizer r and the kernel $k_{\mathbf{d}}$ can be any general differentiable functions of $\mathbf{d}$ with continuous derivatives. In our approach, five base kernels are used, one for each of the five feature sets (LBP, HoG, Gabor, GIST, and Color). The kernel function $k_{\mathbf{d}}$ is selected as a linear combination from the five base kernels: $k_{\mathbf{d}}(\mathbf{x}_u,\mathbf{x}_v) = \sum_{i=1}^{5} d_i k_i(\mathbf{x}_u,\mathbf{x}_v)$ and L2 regularization function is used for $r(\mathbf{d})$. Gradient descent is used for solving 8 using the same algorithm in [31].

4 Experiments

In this section, the effectiveness of our proposed approach for visual comparison of images is tested against state-of-the-art approaches as well as a set of informative baselines. Our experiments are conducted on three public benchmark datasets for image comparison. We find that our proposed approach outperforms the state-of-the-art with a significant improvement in runtime efficiency.

4.1 Evaluation Datasets

Following prior work, evaluation experiments are conducted on three datasets: UT-Zap50K [39], Public Figures dataset (PubFig) [15], and Outdoor Scene Recognition (OSR) [23].

UT-Zap50K Dataset: The UT-Zappos50K introduced in [39] specifically targets the fine-grained attribute comparison task. The dataset is fine-grained as it focuses on a narrow domain of content in the context of an online shopping task with 50,000 catalog shoe images

from Zappos.com. The dataset contains two collections: UT-Zap50K-1 and UT-Zap50K-2. UT-Zap50K-1 is suitable for the task of coarse grained comparison, while the UT-Zap50K-2 includes more challenging images making it more suitable for fine-grained visual comparisons.

PubFig Dataset: The public figures dataset introduced by [15] is one of the largest datasets for face verification and recognition. The same subset of images is used following prior work on attribute comparison [18, 24, 39]. The subset includes 772 images and 11 visual attributes.

OSR Dataset: The Outdoor Scene Recognition (OSR) dataset [23] contains 2688 images from 8 outdoor scene categories and 6 visual attributes covering a large variety of outdoor places. The same set of attributes in [17, 24, 39] is used for evaluation.

4.2 Experimental Settings

Multiple Kernel Learning: For multiple kernel learning, a kernel is computed for each set of features (i.e. five kernels are computed for LBP, HoG, GIST, Gabor, and Color histogram features). A single kernel is computed as a linear combination from the separate kernels for each feature set. Optimal weights for the linear combination are learned using multiple kernel learning as described in 3. We followed the guidelines in [12] for selecting suitable kernels for each feature set. Since the number of features are greater than the number of training samples in LBP, HoG, and Gabor features, a linear kernel is selected for these features since it is less likely to overfit. Experiments using non-linear kernels for these feature sets lead to similar results. This is different for GIST and Color histogram features where the number of features is less than the number of training samples. Exponential χ^2-kernels are used for these feature sets as they have reported state-of-the-art results for measuring the similarity between histograms [29, 42]. The regularization parameter C for Rank-SVM and the χ^2-kernel parameter γ are selected using 10-fold cross-validation.

Experiments show that none of the feature descriptors have the same discriminative power for all the different visual attributes. For example, GIST features perform well when comparing outdoor scenes, whereas local binary patterns performs better when comparing facial images. Therefore, instead of using a single feature set for all visual attributes, it is better to optimally fuse a set of diverse and complementary features - such as features based on color, shape and texture information - in order to better describe the different visual attributes.

Evaluation Metrics: As an evaluation metric, accuracy is reported in terms of the percentage of correctly ordered pairs, following prior work. The time required for comparing test pairs is also reported to illustrate the significant improvement in runtime over prior work. The exact same train/test splits as [17, 24, 39] are used.

Comparisons and Baselines: Our approach is compared against the state-of-the-art as well as some informative baselines. The state-of-the-art approach proposed by Yu and Grauman [39] (FG-LocalPair) is briefly summarized below. In this approach, a local learning model is used for fine-grained visual comparison. Given a novel pair of images, a local ranking model is learned on the fly, using only analogous training pairs. When identifying analogous pairs, a Mahalanobis distance matrix is learned using the Information-theoretic Metric Learning algorithm (ITML) [8]. The selected pairs are used for training a linear ranking model, and the learned model is used for evaluating the test pair.

For a fair comparison with our approach, none of the test images are used for metric learning when implementing the FG-LocalPair model. This is different from the implemen-

tation of [39] where a subset from the test images is used for metric learning. Experiments showed that using this subset of test images in the training pipeline led to a bias in the reported results. For the Pubfig and OSR dataset, the test pairs are removed from the training pipeline and the results are reported in Table 1 and Figure 3. For the UT-Zappos50K dataset, the results on the UT- Zappos50K-1 are not biased as none of the test images were used in the training pipeline. For the UT-Zappos50K-2, more that 90% of the test images were used in the training pipeline. Computing the Mahalanobis distance requires providing two sets for the learner: a set of different image pairs, and a set of similar images. Since the majority of the image pairs labelled to be similar were used for testing in the UT-Zappos50K-2 dataset, we were unable to evaluate the Mahalanobis distance without using samples from the testing data.[1]

A set of baselines is also used for demonstrating the effectiveness of feature fusion and multiple kernel learning. Results from multiple kernel learning are compared to the results obtained using each feature set separately as well as the result from concatenating all the features together instead of using multiple kernel learning (each feature set is normalized separately such that the feature values are between zero and one). The first baseline demonstrates the effectiveness of using a set of features instead of using only one feature set. The second baseline demonstrates the effectiveness of using multiple kernel learning for selecting the optimal feature fusion. Results are also compared to the global non-linear ranking model proposed in [17] (RelTree), which learns a hierarchy of functions, each trained with successively smaller subsets of the data. Code is not available, so the authors' published numbers are reported (available for OSR and PubFig). Finally, our approach is compared to the "LocalPair" approach from [39] which is similar to the FG-LocalPair approach except that the Euclidean distance is used for selecting the neighbors instead of using the Mahalanobis distance.

4.3 Results on Zappos50K

Figure 2 shows the accuracy on UT-Zappos-50K. The same train/test splits in [39] are used. The UT-Zappos-50K-1 is a better representative for the general performance of the model since the average of ten different training and test splits is considered, not just a single split as in UT-Zappos-50K-2. For UT-Zappos-50K-1, our approach outperforms the state-of-the-art for all the four attributes. It is also clear that combining several features instead of considering a single feature is much more effective. The importance of multiple kernel learning is clear, instead of concatenating all the feature descriptors together, multiple kernel learning learns the optimal weights for fusing the features, thus leading to a better performance. For UT-Zappos-50K-2, our approach outperforms the state-of-the-art in three out of the four visual attributes. As explained in the previous section, unbiased results for the FG-LocalPair approach are not available. However, our approach still outperforms the FG-LocalPair results reported in [39] for almost all of the attributes.

4.4 Results on Pubfig

Table 1 shows the accuracy on the PubFig dataset. Image pairs for this dataset originate from category-wise comparisons. On average, 20,000 training pairs are used for training and the evaluation is done on $120,000$ test pairs. Our method outperforms all the baselines.

[1] We'd like to thank the authors for sharing the metric learning code with us. Including test images for metric learning was confirmed via personal communication [5].

	Open	Pointy	Sporty	Comfort
LocalPair [31]	88.53	88.87	92.20	90.90
FG-LocalPair [31]	90.67	90.83	92.67	92.37
LBP Features	90.57	90.53	93.17	91.83
HoG Features	89.73	88.37	89.27	89.67
Gabor Features	87.50	87.13	88.70	87.50
GIST Features	91.77	92.13	93.90	93.10
Color Features	70.60	74.40	79.37	71.70
Concatenation	92.97	92.03	94.63	92.27
Our Approach	**93.63**	**92.57**	**95.07**	**93.20**

(a) Results on the UT-Zappos-50K-1 dataset.

	Open	Pointy	Sporty	Comfort
LocalPair [31]	**71.64**	59.56	61.22	59.75
LBP Features	66.18	65.53	64.08	66.70
HoG Features	62.82	65.19	63.07	66.09
Gabor Features	62.82	60.15	62.79	60.98
GIST Features	62.00	64.42	66.76	63.43
Color Features	46.18	52.47	55.22	57.30
Concatenation	64.36	65.87	67.31	**67.01**
Our Approach	64.91	**66.72**	**67.31**	66.50

(b) Results on the UT-Zappos-50K-2 dataset.

Figure 2: Results on the UT-Zappos-50K dataset.

	Male	White	Young	Smiling	Chubby	Forehead	Eyebrow	Eye	Nose	Lip	Face
RelTree [17]	85.33	82.59	84.41	83.36	78.97	88.83	81.84	83.15	80.43	81.87	86.31
LocalPair [31]	81.53	77.13	83.53	82.60	78.70	89.40	80.63	82.40	78.17	79.77	82.13
FG-LocalPair [31]	86.94	82.89	84.84	83.83	82.84	89.20	85.04	84.87	84.12	84.21	85.74
GIST Features	86.95	83.04	85.80	84.52	81.98	90.75	83.97	84.77	83.17	84.40	86.25
Color Features	62.22	61.27	59.11	57.98	54.73	69.52	60.80	59.79	57.81	57.46	56.94
Gabor Features	80.02	76.33	80.73	80.12	79.23	86.32	76.33	77.44	77.83	79.40	80.64
HoG Features	81.83	77.57	82.28	81.98	80.97	89.68	80.07	80.15	81.87	82.72	83.72
LBP Features	85.83	81.23	86.28	85.74	83.84	92.03	84.44	84.38	85.70	84.09	86.16
Feature Concatenation	87.16	85.21	87.91	86.59	85.96	93.83	84.98	86.90	87.54	86.40	87.74
Our Approach	**87.88**	**85.85**	**88.44**	**87.39**	**86.32**	**93.97**	**85.98**	**87.91**	**88.33**	**86.83**	**88.16**

Table 1: Accuracy comparison for the PubFig dataset.

Most notably, it outperforms the RelTree [17] non-linear approach and the fine-grained local learning approach [39], thus demonstrating the importance of feature fusion and multiple kernel learning.

4.5 Results on OSR

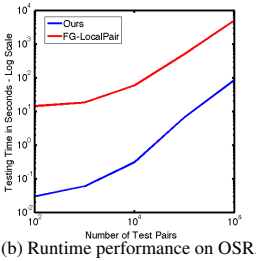
Figure 3 shows the accuracy on the OSR dataset. The OSR dataset is similar to the Pubfig dataset in that supervision data is based on category-wise comparisons. The OSR offers the largest number of training and testing pairs (20,000 pairs for training and 1,800,000 testing pairs on average), hence, runtime performance evaluation has been performed on this dataset. In terms of accuracy, the performance is similar to other datasets where our approach outperforms the state-of-the-art results as well as the baselines. One advantage of our approach is that a single ranking model is learned instead of learning a new model for each test query. Figure 3-(b) shows the runtime performance for our proposed approach as compared to the FG-LocalPair approach [39]. The testing time in seconds is plotted using a logarithmic scale. The machine used for evaluating the runtime performance has the following configuration: 16 GB 1600 MHz DDR3 memory and 2.8 GHz Intel quad-core Core i7 processor. The improvement in runtime is obvious, while our approach evaluates a million test pairs in around one minute, the local learning approach requires more than an hour to produce the results.

5 Conclusion

In this paper, the standard multiple kernel learning formulation is extended to the learning to rank problem. Effectiveness of the proposed approach is demonstrated on the visual image comparison task. Although MKL has been extensively used for object recognition and image categorization, this is the first time it has been used for image comparison. Through extensive experiments, the advantage of our approach is clearly demonstrated both in terms of accuracy

	Natrl	Open	Persp.	LgSize	Diag	ClsDepth
RelTree [□]	95.24	92.39	87.58	88.34	89.34	89.54
LocalPair [□]	94.63	93.27	88.33	89.40	90.70	89.53
FG-LocalPair [□]	94.68	92.90	88.03	88.84	89.65	89.91
GIST Features	94.62	91.44	85.66	86.47	86.31	86.80
Gabor Features	74.16	79.80	72.20	71.88	69.90	74.27
HoG Features	91.73	90.37	84.47	84.79	84.58	82.14
LBP Features	94.26	89.41	86.15	86.18	87.13	88.14
Feature Concatenation	95.54	91.77	88.40	88.54	89.56	90.02
Our Approach	**96.20**	**93.69**	**89.84**	**90.06**	**91.11**	**91.54**

(a) Accuracy comparison for the OSR dataset.



(b) Runtime performance on OSR.

Figure 3: Accuary and runtime performance on the OSR dataset.

and runtime efficiency. Future work includes exploring more applications of multiple kernel learning for ranking, such as web search and recommendation systems.

References

[1] Francis R Bach, Gert RG Lanckriet, and Michael I Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *Proceedings of the twenty-first international conference on Machine learning*, page 6. ACM, 2004.

[2] Arijit Biswas and Devi Parikh. Simultaneous active learning of classifiers & attributes via relative feedback. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 644–651. IEEE, 2013.

[3] Serhat S Bucak, Rong Jin, and Anil K Jain. Multiple kernel learning for visual object recognition: A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(7):1354–1369, 2014.

[4] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96. ACM, 2005.

[5] JunKai Chen, Zenghai Chen, Zheru Chi, and Hong Fu. Emotion recognition in the wild with feature fusion and multiple kernel learning. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 508–513. ACM, 2014.

[6] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20 (3):273–297, 1995.

[7] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[8] Jason V Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, pages 209–216. ACM, 2007.

[9] Yoav Freund, Raj Iyer, Robert E Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *The Journal of machine learning research*, 4: 933–969, 2003.

[10] Mehmet Gönen and Ethem Alpaydın. Multiple kernel learning algorithms. *The Journal of Machine Learning Research*, 12:2211–2268, 2011.

[11] Ralf Herbrich, Thore Graepel, and Klaus Obermayer. Large margin rank boundaries for ordinal regression. *Advances in neural information processing systems*, pages 115–132, 1999.

[12] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. A practical guide to support vector classification, 2003.

[13] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM, 2002.

[14] Adriana Kovashka, Devi Parikh, and Kristen Grauman. Whittlesearch: Interactive image search with relative attribute feedback. *International Journal of Computer Vision*, pages 1–26, 2015.

[15] Neeraj Kumar, Alexander C Berg, Peter N Belhumeur, and Shree K Nayar. Attribute and simile classifiers for face verification. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 365–372. IEEE, 2009.

[16] Tzu-Ming Kuo, Ching-Pei Lee, and Chih-Jen Lin. Large-scale kernel ranksvm.

[17] Shaoxin Li, Shiguang Shan, and Xilin Chen. Relative forest for attribute prediction. In *Computer Vision–ACCV 2012*, pages 316–327. Springer, 2013.

[18] Lucy Liang and Kristen Grauman. Beyond comparing image pairs: Setwise active learning for relative attributes. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 208–215. IEEE, 2014.

[19] Yen-Yu Lin, Tyng-Luh Liu, and Chiou-Shann Fuh. Dimensionality reduction for data in multiple feature representations. In *Advances in Neural Information Processing Systems*, pages 961–968, 2009.

[20] Yen-Yu Lin, Tyng-Luh Liu, and Chiou-Shann Fuh. Multiple kernel learning for dimensionality reduction. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(6):1147–1160, 2011.

[21] Bangalore S Manjunath and Wei-Ying Ma. Texture features for browsing and retrieval of image data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18 (8):837–842, 1996.

[22] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):971–987, 2002.

[23] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3): 145–175, 2001.

[24] Devi Parikh and Kristen Grauman. Relative attributes. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 503–510. IEEE, 2011.

[25] Amar Parkash and Devi Parikh. Attributes for classifier feedback. In *Computer Vision–ECCV 2012*, pages 354–368. Springer, 2012.

[26] Daniel A Reid and Mark S Nixon. Using comparative human descriptions for soft biometrics. In *Biometrics (IJCB), 2011 International Joint Conference on*, pages 1–6. IEEE, 2011.

[27] David Sculley. Combined regression and ranking. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 979–988. ACM, 2010.

[28] Sören Sonnenburg, Gunnar Rätsch, Christin Schäfer, and Bernhard Schölkopf. Large scale multiple kernel learning. *The Journal of Machine Learning Research*, 7:1531–1565, 2006.

[29] V Sreekanth, Andrea Vedaldi, Andrew Zisserman, and C Jawahar. Generalized rbf feature maps for efficient detection. 2010.

[30] Devis Tuia, Gustavo Camps-Valls, Giona Matasci, and Mikhail Kanevski. Learning relevant image features with multiple-kernel classification. *Geoscience and Remote Sensing, IEEE Transactions on*, 48(10):3780–3791, 2010.

[31] Manik Varma and Bodla Rakesh Babu. More generality in efficient multiple kernel learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1065–1072. ACM, 2009.

[32] Raviteja Vemulapalli, Jaishanker K Pillai, and Rama Chellappa. Kernel learning for extrinsic classification of manifold features. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 1782–1789. IEEE, 2013.

[33] Jiang Wang, Yang Song, Tommy Leung, Catherine Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1386–1393. IEEE, 2014.

[34] Qiang Wu, Chris JC Burges, Krysta M Svore, and Jianfeng Gao. Ranking, boosting, and model adaptation. Technical report, Technical report, Microsoft Research, 2008.

[35] Jun Xu and Hang Li. Adarank: a boosting algorithm for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 391–398. ACM, 2007.

[36] Zhongwen Xu, Ivor W Tsang, Yi Yang, Zhigang Ma, and Alexander G Hauptmann. Event detection using multi-level relevance labels and multiple features. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 97–104. IEEE, 2014.

[37] Yi-Ren Yeh, Ting-Chu Lin, Yung-Yu Chung, and Y-CF Wang. A novel multiple kernel learning framework for heterogeneous feature fusion and variable selection. *Multimedia, IEEE Transactions on*, 14(3):563–574, 2012.

[38] Aron Yu. Private communication, 2015.

[39] Aron Yu and Kristen Grauman. Fine-grained visual comparisons with local learning. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 192–199. IEEE, 2014.

[40] Hwanjo Yu, Youngdae Kim, and Seungwon Hwang. Rv-svm: An efficient method for learning ranking svm. In *Advances in Knowledge Discovery and Data Mining*, pages 426–438. Springer, 2009.

[41] Shi Yu, Tillmann Falck, Anneleen Daemen, Leon-Charles Tranchevent, Johan AK Suykens, Bart De Moor, and Yves Moreau. L2-norm multiple kernel learning and its application to biomedical data fusion. *BMC bioinformatics*, 11(1):309, 2010.

[42] Jianguo Zhang, Marcin Marszałek, Svetlana Lazebnik, and Cordelia Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International journal of computer vision*, 73(2):213–238, 2007.

[43] Zhaohui Zheng, Keke Chen, Gordon Sun, and Hongyuan Zha. A regression framework for learning ranking functions using relative relevance judgments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 287–294. ACM, 2007.

[44] Alexander Zien and Cheng Soon Ong. Multiclass multiple kernel learning. In *Proceedings of the 24th international conference on Machine learning*, pages 1191–1198. ACM, 2007.