Semi-Supervised Learning of Object Categories from Paired Local Features

Wen Wu and Jie Yang
School of Computer Science, Carnegie Mellon University
5000 Forbes Ave, Pittsburgh, PA 15213, U.S.
{wenwu,jie.yang}@cs.cmu.edu

ABSTRACT

This paper presents a semi-supervised learning (SSL) approach to find similarities of images using statistics of local matches. SSL algorithms are well known for leveraging a large amount of unlabeled data as well as a small amount of labeled data to boost classification performance. Our approach proposes to formulate the problem of matching two images as an SSL based classification problem of image pairs with a minimal amount of labeled pairs. We apply a Gaussian random field model to represent each image pair as vertices in a weighted graph and the optimal configuration of the field is obtained by harmonic energy minimization. A symmetrical feature selection criterion is first introduced to select robust matches of local keypoints between two images. The Mallows distance is then adopted to combine multiple cues from statistics of local matches. Our experiments confirm that our SSL based approach not only boost classification performance but also improve robustness of the learned category model using only simple local keypoint features.

Categories and Subject Descriptors

I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis: Object recognition

General Terms

Algorithms

Keywords

Semi-Supervised Learning, Object Classification

1. INTRODUCTION

The rapid growth of digital images on the Internet has created many new challenges for image index and retrieval research. However, a vast amount of available image data also brings opportunities for researchers to solve some well-studied problems from new perspectives. This observation motivates us to take advantage of a large amount of image data (unlabeled) from public datasets or Internet to re-study one of classical image retrieval problems, namely comparing two images (or image regions) to determine if they contain the same kind of dominant objects.

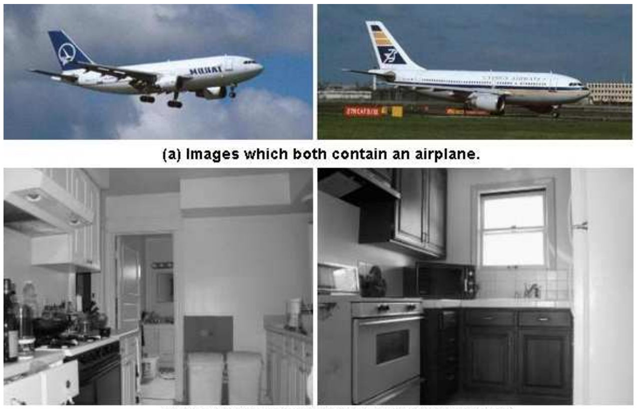

(a) Images which both contain an airplane.
(b) Images which both describe kitchen scene.

Figure 1: Two kinds of image matching: object based matching (a) and scene based matching (b). Our approach focuses on the first case but also presents experiments on the second case.

Similarity between two images can be interpreted in different senses. For example, two images both describing kitchen scene can be considered as similar or relevant even though they capture two different kitchens. However, in this paper, we adopt another definition of image similarity. Two images (or image regions) are considered to be similar *if and only if the dominant objects in two images are the same (kind of) objects*. Fig.1 illustrates these two cases. If an image contains multiple objects, we assume that a preprocessing step can be applied to segment the image into regions that contain only one dominant object within each region, and then apply the proposed method. Image similarity is usually measured based on a certain criterion through an image matching process. Our task is, in fact, an image matching problem.

The image matching problem has been popular since the beginning of image retrieval research and the corner stone of many different practical applications [5, 21, 4]. While research has been particularly focused on apply different kinds of image features and knowledge to boost matching

CIVR'08, July 7–9, 2008, Niagara Falls, Ontario, Canada.

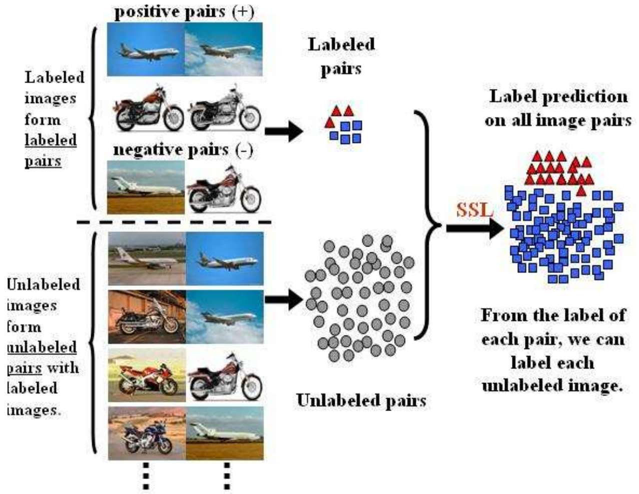

positive pairs (+)
Labeled images form labeled pairs
negative pairs (-)
Labeled pairs
Label prediction on all image pairs
SSL
Unlabeled images form unlabeled pairs with labeled images.
Unlabeled pairs
From the label of each pair, we can label each unlabeled image.

Figure 2: Semi-supervised learning (SSL) of the *airplane* and *motorbike* categories. Triangles (△) indicate matched image pairs (positive), boxes (□) indicate unmatched pairs (negative) and circles (○) represent unlabeled image pairs that include one labeled image and one unlabeled image.

performance [24, 19], little attention has been paid on using a large amount of unlabeled images together with a small amount of labeled data (matched and unmatched image pairs) to learn a robust image matcher. On the other hand, semi-supervised learning (SSL), which utilizes the availability of a large amount of unlabeled data to enhance classification performance, has evolved to a full fledged research body in the last decade [1]. Because the SSL requires less human effort and can provide higher accuracy, it is of great interest both in theory and practice and has been applied in many areas [10, 18], e.g., semantic video retrieval [3].

This paper presents an SSL based approach to match images using statistics of local matches. Our approach proposes formulating the problem of matching two images as an SSL based classification problem of image pairs solved by a Gaussian random field. Our approach is different from the previous research in three aspects, *the problem formulation*, *the feature representation* and *the matching model*, all of which will be described in detail in next two sections. Here is a brief summary of the related work. Zhou et al. proposed to combine SSL and active learning to exploit unlabeled data to improve the performance of content-based image retrieval [28]. Kuck et al. attempted to interpret a class of data association tasks (e.g., object recognition and semantic concept detection) as a constrained semi-supervised learning problem [10]. Tian et al. provided an analysis on the value of unlabeled data by considering different distributions of the labeled and unlabeled data and showing the migrating effect for semi-supervised within the CBIR context [23]. Other SSL based work in image and video retrieval include [17]. Hertz et al. also introduced a technique that uses binary relations between images to learn constraints for image retrieval [8]. Grauman and Darrell applied a graph-based technique for unsupervised category learning using correspondence patterns over local features [7]. Lazebnik et al. proposed a method for recognizing *scene* categories based on approximate global geometric correspondence [11]. Our work is close to [2, 9] where matching is formulated as probabilistic relaxation. On the other hand, interest points have been actively studied and applied to image and video retrieval applications [20, 27, 16, 22], which inspired our feature representation using only local keypoints in this task.

The reminder of the article is organized as follows. Section 2 describes the semi-supervised image matching problem formulation. Section 3 lays out the steps to compute the pairwise distance between two image pairs from only local keypoints extracted from images. Section 4 introduces the Gaussian random field framework and harmonic functions which serve as the learning engine of our SSL method. Section 5 show the experimental results and analysis. Finally, we summarize our findings in Section 6.

2. PROBLEM FORMULATION

The goal of this research is to address the problem of matching two images which contain similar dominant objects by exploiting the intrinsic similarity structure of the data through combining both unlabeled and labeled image pairs within an SSL framework. We call the new approach semi-supervised image matching (SIM). Figure 2 depicts the proposed semi-supervised learning concept using the *airplane* and *motorbike* categories. Triangles indicate matched image pairs (△), squares indicate unmatched pairs (□) and circles represent unlabeled image pairs (○).

Formally, we formulate SIM as follows. Given an image set, $X = \{x_1, ..., x_l, \, x_{l+1}, ..., x_n\}$, and a label set, $\mathcal{C} = \{1, ..., c\}$, the first l images have labels $\{y_1, ..., y_l\} \in \mathcal{C}$ and the remaining images are unlabeled. We call $L = \{(x_1, y_1), ..., (x_l, y_l)\}$ the labeled image set and $U = \{x_{l+1}, ..., x_n\}$ the unlabeled image set. The goal is to predict labels on U. From L we build labeled image pair set L_p in the following ways. For x_i, x_j when $y_i = y_j$, create a pair of x_i, x_j and label it as positive (+) since x_i and x_j belong to the same class. Also for each x_i randomly select K other images where $y_j \neq y_i, j = 1, ..., K$, create K pairs from them and label them as negative (-). Thus L_p set consists of positive image pairs and negative ones. Denote pairs in L_p as Z_i^j. Set K to be at most $c - 1$ (c is the number of classes in L), the size of L_p is $O(lc)$. Similarly, from U and L together we create the unlabeled image pair set U_p. For each unlabeled image, $x_p, p = (l + 1), ..., n$, randomly select one image per class in L, x_q, where $y_q = 1, ..., c$ (each of which belongs to one class), and build c pairs. Denote these pairs as $Z_p^q, q = 1, ..., c$. Since x_p is an unlabeled image, Z_p^q is put in U_p as an unlabeled pair. Thus the size of U_p is $(n - l)c$, in total there are $O(nc)$ image pairs in L_p and U_p.

Based on L_p and U_p we build a connected graph, $G = (V, E)$, with nodes V corresponding to nc data points and the edges, E, are weighted by an $nc \times nc$ affinity matrix, W, computed using the Mallows distance. Essentially, we transform image matching into a binary classification problem, where the positive label (+) indicates two images are from the same class while the negative label (-) means two images do not belong to the same class. Once labels are predicted on all unlabeled image pairs, the label of each unlabeled image can be inferred based on pairwise relationship between this unlabeled image and c labeled images. For example, if Z_p^q is labeled as positive (+), x_p is labeled as the class of x_q (note x_p is originally an unlabeled image from U and x_q is a labeled image from L). Thus, multi-label learning is naturally

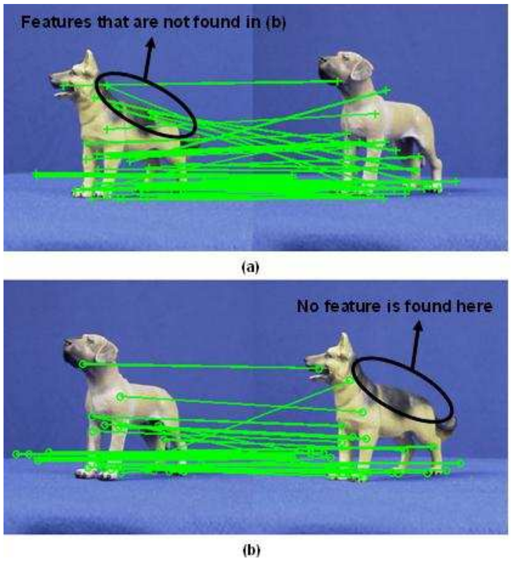


Figure 3: Asymmetric matchings obtained by Lowe's matching criterion [12]. Notice features selected on one dog's back in (a) are selected in (b).

handled by our framework. The computational complexity of the new problem on L_p and U_p ($O(n^2c^2)$) is higher than the original SSL on L and U ($O(n^2)$). But the advantage of exploring the more computational complex problem is obvious - extracting informative features from a pair of images instead of a single image.

3. LOCAL FEATURES TO IMAGE-PAIR DISTANCES

In this section, we describe how SIM represents the feature to compare two image pairs. Detecting interesting local features from different images has been shown to useful for a diverse class of computer vision problems [12]. Give two images x_i and x_j, we first extract Harris-affine and Hessian-affine keypoints [14] and compute their SIFT features. Next a symmetrical matching algorithm is applied to select symmetrical matches of local features between these two images. Once the local matches are available, our method computes displacement and orientation statistics of local feature matching lines. Finally, to measure the similarity distance between two pairs, Z_i^j and Z_p^q, SIM calculates the sum of squared Mallows distances between two image pairs.

In next, we first describe the criterion to select symmetrical matches of local features, and then, computation of displacement and orientation statistics of matching lines, and finally, the function to combine local matching patterns to obtain pairwise distances among image pairs.

3.1 Symmetrical Match of Local Features

Accurately matching extracted local features is crucial to ensure the quality of high-level applications such as object detection and recognition. The key issue is to define a reliable matching criterion so that correct matched candidates are not missed while mismatching caused by a background clutter or a noisy environment. Many state-of-the-art descriptor matching algorithms choose to use the criterion proposed by Lowe [12], which is defined as a threshold on the ratio of distance from the closest neighbor to that of the second-closest neighbor. The criterion has been proved to be reliable and robust in many applications [12]. However, there is an asymmetric phenomenon which often happens when we apply Lowe's matching algorithm on two images twice with different matching directions, one from x_1 to x_2, and the other from x_2 to x_1. The images in Figure 3 shows the matching results by Lowe's method. As we can see, the matched features from (a) and (b) are not identical for the same two images. The reason actually roots from the definition of the matching ratio Lowe used. Assume a feature f_i in x_1, we find its best match f_j from x_2. However, in the other direction, for the feature f_j in x_2, f_i is not guaranteed to be f_j's best match in x_1 by Lowe's matching criterion. This observation motivates us to put a symmetrical constraint on Lowe's matching criterion, which means, only when both f_i and f_j are the best match to each other from both matching directions using Lowe's method, the pairs will then be selected and paired.

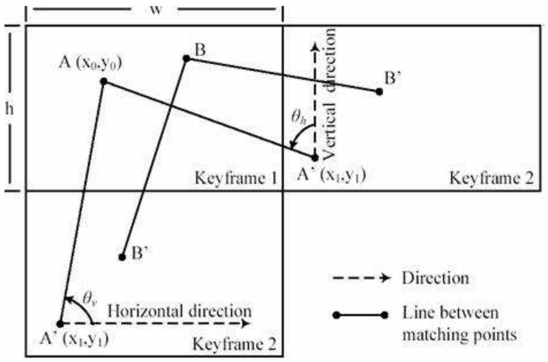


Figure 4: Computing horizontal and vertical angles from local feature matchings of two images.

3.2 Statistics of Local Matchings

Once we have a set of good local feature matches between x_i and x_j, we compute the orientation patterns of feature matches by putting x_i and x_j side by side horizontally and vertically. We also compute the displacement changes of matched descriptors on the horizontal and vertical directions by normalizing the sizes of x_i and x_j to be identical and putting x_i on the top of x_j. Thus we capture the matching patterns of this pair Z_i^j with four histograms, namely, two orientations and two displacements, denoted by $\mathbb{H} = \{H_o^1, H_o^2, H_d^1, H_d^2\}$. H_o^1 and H_o^2 are estimated by aligning x_i and x_j horizontally and vertically. Depending on the alignment, a histogram (36 bins with a step 5 degree from 0 to π) is composed of the quantized angles formed by the descriptor matching lines with respect to the horizontal or vertical axis. The vertical angle, θ_v, is computed as,

$$\theta_v = arccos(\frac{x_2 - x_1}{\sqrt{(x_2 - x_1)^2 + (y_2 + h - y_1)^2}}), \qquad (1)$$

as shown in Figure 4. Similarly, the horizontal orientation angle, θ_h, is computed as $\theta_h = arccos(\frac{y_2 - y_1}{\sqrt{(x_2 + w - x_1)^2 + (y_2 - y_1)^2}})$.

The horizontal displacement is defined as $d_h = x_2 - x_1$. The $sign(d_h) > 0$ indicates the keypoint moves to the right from I_1 to I_2, otherwise to the left. The vertical displacement is defined as $d_v = y_2 - y_1$. H_o^1 is the histogram on θ_v,H_o^2 is the histogram on θ_h, H_d^1 is on d_h and H_d^2 is the histogram on d_v. Orientation statistics are shown useful to the near-duplicate keyframe tracking problem [15].

3.3 Image-Pair Distances

Denote the image pair space as Θ. Given two image pairs, Z_i^j and Z_p^q from Θ, we combine the orientation and displacement matching patterns to obtain multiple distributions to characterize them (Figure 5). We use the Mallows distance to compute the distance $D(Z_i^j, Z_p^q)$ between two image pairs Z_i^j and Z_p^q [13]. Assume a random variable $F \in R^b$ follows the distribution λ_1 and $G \in R^b$ follows λ_2. Let $\Phi(\lambda_1, \lambda_2)$ be the set of joint distributions over F and G with marginal distributions of F and G constrained to λ_1 and λ_2 respectively. Especially, if $\zeta \in \Phi(\lambda_1, \lambda_2)$, then ζ has sample space $R^b \times R^b$ and its marginal probabilities $\zeta_F = \lambda_1$ and $\zeta_G = \lambda_2$. The Mallows distance is defined as the minimum expected distance between F and G optimized over all joint distributions $\zeta \in \Phi(\lambda_1, \lambda_2)$ as follows,

$$D(\lambda_1, \lambda_2) \triangleq \arg \min_{\zeta \in \Phi(\lambda_1, \lambda_2)} (\mathbb{E} \parallel F - G \parallel^p)^{\frac{1}{p}}, \qquad (2)$$

where $\parallel \cdot \parallel$ denotes the L_p distance between two vectors. In our work, we use the L_2 distance. For the discrete distributions such as histograms, $\mathbb{H} = \{H_o^1, H_o^2, H_d^1, H_d^2\}$ which we compute from above feature extraction steps, the optimization involved in computing the Mallows distance can be solved by linear programming. Let two discrete distributions from $\mathbb{H}$ be λ_1, λ_2 and λ_1, λ_2 are within the same channel out of four, for example, both are horizontal orientation histograms.

$$\lambda_i = \{(z_i^1, q_i^1), (z_i^2, q_i^2), ..., (z_i^{n_i}, q_i^{n_i})\}, i = 1, 2; \qquad (3)$$

where $n_1 = n_2$, z_i^j is the j-th element of z_i and q_i^j is the associated probability, and $n_i, i = 1, 2$ is the vector dimension. Therefore, Eq.(9) leads to the following optimization problem,

$$D^2(\lambda_1, \lambda_2) = \arg \min_{w_{i,j}} \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} w_{i,j} \parallel z_1^i - z_2^j \parallel^2, \qquad (4)$$

subject to $\sum_{j=1}^{m_2} w_{i,j} = q_1^i, i = 1, ..., m_1, \sum_{i=1}^{m_1} w_{i,j} = q_2^j, j = 1, ..., m_2$, and $w_{i,j} \geqq 0, i = 1, ..., m_1, j = 1, ..., m_2$. The optimization problem indicates that the squared Mallows distance is a weighted sum of pairwise squared L_2 distances between any support vector of λ_1 and any of λ_2. With an objective to minimize the aggregated distance, the optimization is performed over the matching weights between support vectors in the two distributions. The weights $w_{i,j}$ are constrained to be positive or zero and q_i^j determines the amount of influence from z_i^j on the overall distribution distance.

Since data point Z_i^j is each characterized by four histograms (normalized along each variable dimension), aka discrete distributions, SIM measures their pairwise distances by the sum of squared Mallows distances between individual histograms. In order to simplify the notation of Z_i^j in the following definitions, we denote Z_i^j as $\delta_k, k = 1, 2, ..., K$ where each k uniquely matches a particular i and j. Denote the distance by $\mathbb{D}(\delta_i, \delta_j), \delta_i, \delta_j \in \Theta$, and then the global distance between δ_i and δ_j is defined as,

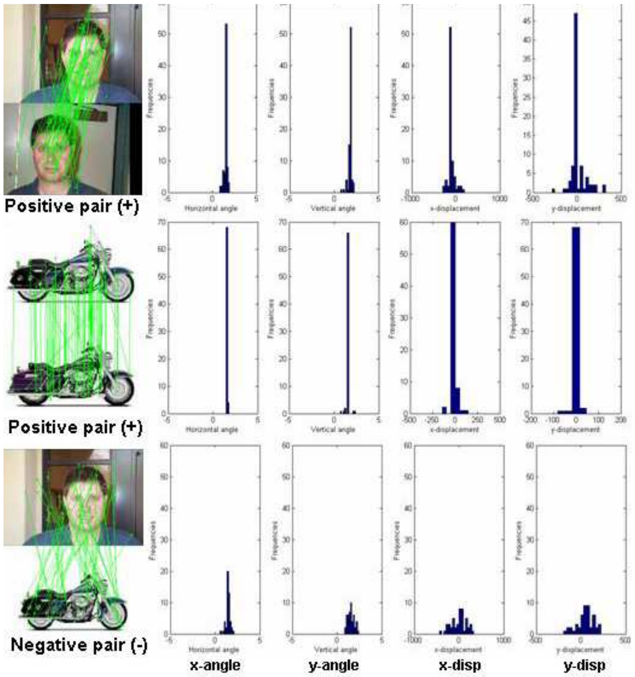


Figure 5: Examples of positive and negative image pairs and their orientation and displacement statistics. Notice similarity between positive pairs and difference between positive and negative.

$$\mathbb{D}(\delta_i, \delta_j) \triangleq \sum_{l=1}^{d} D^2(\delta_{i,l}, \delta_{j,l}), \qquad (5)$$

where d is the super-dimension of Θ.

4. GAUSSIAN RANDOM FIELD (GRF)

Since we can measure a distance between any two image pairs, we represent all image pairs as vertices in a weighted graph with edge weights representing the pairwise image-pair distances. We then adopt GRF as the SSL engine in our approach and the solution can be efficiently obtained using matrix methods or belief propagation.

GRF is one of graph-based SSL algorithms [29]. The goal of GRF is to first compute a real-valued labeling function, $h(\cdot) : V- > \mathcal{R}$, on G with certain nice properties, and then to assign labels for U based on $h(\cdot)$. The labeling function is constrained to assign labels such as $h(i) = h_l(i) \equiv y_i$, on L, $i = 1, ..., l$. The aim of the Gaussian field configuration is to make unlabeled points that are nearby in the graph have similar labels. This motivates the choice of quadratic energy function, $E(g) = \frac{1}{2} \sum_{i,j} w_{ij}(h(i) - h(j))^2$. In the n-dim Euclidean space, $x \in \mathbb{R}^n$, the weight matrix, W, can be defined as, $w_{ij} = exp\left(-\sum_{d=1}^{n} \frac{(x_{id} - x_{jd})^2}{2\sigma_d^2}\right)$, where x_{id} is d-th component of the feature vector $x_i \in \mathbb{R}^n$, and $\sigma_1, ..., \sigma_n$ are length scale hyperparameters for each dimension. Therefore, nearby data points in the Euclidean space are assigned large weights. The diagonal of W is set as 1, $w_{ii} = 1$.

In order to assign a probability distribution on $h(\cdot)$, a Gaussian random field is formed as $p_\beta(h) = \frac{e^{-\beta E(h)}}{Z_\beta}$, where β is an ”inverse temperature” parameter and Z_β is the partition function $Z_\beta = \int_{h|h_l=L} exp(-\beta E(h))dh$, which normalizes over all functions constrained to h_l on L, where $(x_i, y_i) \in L, i = 1, ..., l$. To compute the harmonic solution of functions $h(\cdot)$, we minimize the quadratic energy function $E(h)$ subject to the labeled data constraint.

$$h = \arg\min_{h|_L=h_l} E(h) = \arg\min_{h|_L=h_l} \frac{1}{2}\sum_{i,j} w_{ij}(h(i) - h(j))^2, \tag{6}$$

in other words, it satisfies the equation, $\Delta \cdot h = 0$ on all unlabeled points in U and is subject to be equal to h_l on L. Here Δ is called the *combinatorial Laplacian*, and given in matrix form as $\Delta = D - W, whereD = diag(d_i), d_i = \sum_j w_{ij}$.

The harmonic property indicates that the value of h at each unlabeled point is the average of h at its neighborhood points,

$$h(j) = \frac{1}{d_j}\sum_i w_{ij} h(i), \text{ for } j = l + 1, ..., l + u, \tag{7}$$

which maintains the smoothness constraint of $h(\cdot)$ with respect to the graph G. Expressed in the matrix form, Eq(4) can be rewritten as $h(\cdot) = Q \cdot h(\cdot)$, where $Q = D^{-1}W$. Because of the maximum principle of harmonic functions [29], the solution of g is unique and it is either a constant or satisfies $0 < h(j) < 1$ for $j \in U$. To compute the harmonic solution in terms of matrix operations, W is split into 4 blocks in terms of separation of L and U.

$$W_{n\times n} = \begin{bmatrix} W_{ll} & W_{lu} \\ W_{ul} & W_{uu} \end{bmatrix}_{n\times n}. \tag{8}$$

Denote the target labeling function $h = \begin{bmatrix} h_l \\ h_u \end{bmatrix}_{n\times c}$, and $h_l(\cdot)$ denotes labels on L, $h_u(\cdot)$ denotes labels on U, and c is the number of classes. The unique harmonic solution $\Delta \cdot h(\cdot) = 0$ subject to $h(\cdot)|_L = h_l(\cdot) \equiv y_i, i = 1, ..., l$ is given as a $u \times c$ matrix $h_u(\cdot)$.

$$h_u(\cdot) = (D_{uu} - W_{uu})^{-1} W_{ul} \cdot h_l. \tag{9}$$

We can see that GRF has a quadratic loss function with an infinite weight on labeled data, so that label values of the labeled data are fixed, and a regularizer based on the graph combinatorial Laplacian. Notice $h(i) \in R$, which allows for the simple closed-form solution for the node marginal probabilities in Eq.(9). The resulting classification algorithm for GRF can be viewed as a form of nearest neighbor approach, where the nearest labeled points are computed in terms of a random walk on the graph. Some recent research has applied GRF and harmonic functions to a number of computer vision problems. Grady and Funka-Lea applied the harmonic function method to medical image segmentation problems, where a user labels different organs with a few strokes [6]. Wu and Yang applied an iterative version of GRF to semi-automatic object labeling tasks [25]. Fig. 6 shows the semi-supervised image matching (SIM) algorithm.

Input A set of n images, Γ, which leads to the form of its image pair set, Θ, which includes $\frac{n\times(n-1)}{2}$ examples (pairs).

Output Similarity label on each image pair from Θ. ‘+‘ indicates matched pair and ‘−‘ means the unmatched image pairs.

Semi-supervised Image Matching

1. *Separate* Θ *into* L *and* U. Labels are given to examples in L. U is the unlabeled set. $U \gg L$ in size.

2. *Extract local keypoints and descriptors.* Harris-affine and Hessian affine detectors are used to detect local keypoints on each image. SIFT descriptor is used.

3. *Extract symmetrical keypoint pairs.* Given two images, a symmetrical matching criterion is applied to select matched keypoint pairs.

4. *Compute statistics of keypoint pairs.* Both orientation and displacement histograms are computed to characterize the matching patterns between the two images.

5. *Compute the Mallows distance among image pairs.* The pairwise distance in Θ is measured by the sum of squared Mallows distances between individual histograms extracted from individual image pairs.

6. *Gaussian random field learning.* Represent all image pairs as vertices in a weighted graph, G. The predicted labels on unlabeled pairs are obtained by solving harmonic functions on the random field.

Figure 6: Semi-supervised Image Matching (SIM).

5. EXPERIMENTS

In this section, we report results on three different data sets (Figure 8): Caltech-4 [1], Caltech-101 [4] [2] and 15 scene categories [11]. Follow the conditions used by [11], we perform all feature extractions on the grayscale images even when color images are available. All experiments are repeated ten times with different randomly selected images, and we report the mean and standard deviation of per-class accuracy from individual runs.

The first series of experiments was designed to compare the contrastive performance of various classifiers and feature spaces on Caltech-4 data set. Caltech-4 is a common benchmark data set containing four different object classes, which are 1074 images of airplanes from the side, 1155 rear views of cars, 450 frontal face images and 826 images of motorbikes from the side. Part of images from Caltech-4 are also included in Caltech-101 set. Most of images have medium resolution, i.e., smaller than 800*500 pixels. In the first experiment, we split images of each class into halves. The first 50% of the images were used as training data and the rest 50% were used as testing data. For each image pair, two orientation and two displacement histograms were extracted in the feature space, where orientation had a fixed number of

[1] http://www.robots.ox.ac.uk/ vgg/data3.html

[2] http://www.vision.caltech.edu/Image_Datasets/Caltech101

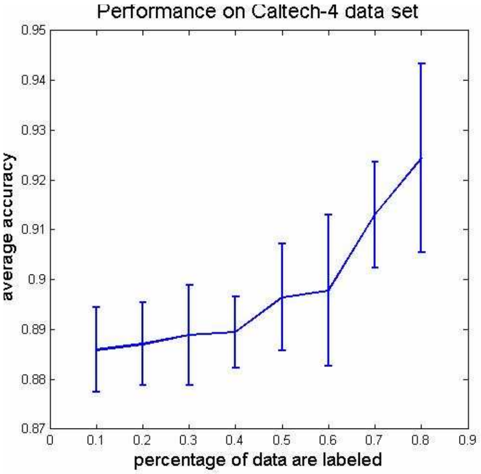


Figure 7: SIM's result on Caltech-4 dataset.

36 bins and displacement had 32 bins. For baseline classifiers' training, all features are concatenated thus we have a total 136 one-dimensional features for *orient+disp* and 72 features for *orient* only.

Table 1 shows the accuracy for the following baseline classifiers: k Nearest Neighbor with L_2 distance(kNN L_2), Linear SVM(LSVM), Radial kernel SVM with $\rho = 0.01$ (RSVM) and χ^2 kernel SVM with $\rho = 0.01$ (SVM χ^2). As shown, SVM χ^2 achieves the best performance followed by RSVM, indicating that the χ^2 distance is the best suited distance metric for supervised learning of object category model. Adding displacement info slightly improves the classification performance. In our following experiments, the χ^2 kernel SVM using *orient+disp* features was chosen as our baseline classifier due to its good performance.

Next, let us examine the behavior of our proposed method given a varying amount of labeled data. Initially, our training data consisted of 10% of images from each class. All the remaining 90% images from the same class were used as unlabeled data. Figure 7 summarizes the accuracy of prediction produced by SIM. For each percentage level tested, we ran the algorithm 10 times, each time with a different random selection of images from 3505 total images with the same percentage of images per class. This result affirms that the more labeled data, the better our method performs. Given 50% of images per class, our method achieves about 90% accuracy, which gains a huge improvement over the baseline and is comparable to the state of the art method on the same dataset [7].

	kNNL_2	**LSVM**	**RSVM**	**SVM χ_2**
orient	50.6±0.4	56.3±0.3	56.9±0.4	59.7±0.4
orient+disp	53.2±0.4	57.2±0.5	58.3±0.3	**61.5**±0.4

Table 1: Accuracy for different classifiers using orientation and orientation plus displacement with 50%/50% training/testing splits on Caltech-4.

	SVMχ_2	**SIM-O**	**SIM**	**Pyd** [11]
minaret	84.8	91.3	97.8	97.6
windsor_chair	80.3	84.5	92.3	94.6
joshus tree	75.8	83.9	88.2	87.9
okapi	71.4	82.3	86.5	87.8
cougar-body	26.4	35.6	42.8	27.6
beaver	0.42	24.8	27.2	27.5
crocodile	18.5	20.1	21.3	25.0
ant	32.8	42.7	46.8	25.0
*Avg.Acc.*101	53.4±0.9	60.3±0.7	64.8±0.7	64.6±0.8

Table 2: A comparison on Caltech101 dataset. SVMχ^2: our baseline; SIM-O: using only orientation statistics; SIM: our method using full feature set. Pyd [11]: pyramid matching with $L = 2, M = 200$.

Our next set of experiments is on Caltech-101. This data set contains 101 categories with from dozens to a few hundred images per class. Most images are medium resolution smaller than 800*800 pixel. Most images have objects at centers and some occupy most of the image. We followed the experiment set up used in [11], namely, used 30 images per class as training data and test on the rest. Table 2 summarizes the comparison among different classifiers. In order to better compare our method with spatial pyramid matching method, we list overall result and individual results on eight categories provided by Lazebnik et al. [11]. The results indicate that utilizing unlabeled data improves the classification performance by comparing SVM and SIM methods. Also, using only orientation statistics can lead good performance but still little worse than using the full feature set. SIM achieves a accuracy rate of 64.8 ± 0.7 which is comparable to 64.6 ± 0.8 by Lazebnik et al [11] and 66.2 ± 0.5 by H. Zhang [26]. On the hard categories (like crocodile and beaver) SIM also performs poorly. We conjecture that the poor performance is due to textureless animal skins, inconsistent animal poses and camouflage animals have in their environments.

Fig.9 shows comparison between Lowe's matching criterion [12] and our symmetrical matching criterion on two outdoor scene images. As we can see, our criterion improves matching performance in the second column and matchings in (b) and (d) are essentially symmetrical.

The last experiment was designed to assess the performance of our method on *scene* classification task. As seen from above, our proposed method works very well on object based image classification tasks. However, our approach was not designed to deal with classifying *scene* images. Our final set of experiments is on the 15 scene categories dataset [4]. As seen in Figure 8, this dataset has 15 scene classes and images are all in grayscale. Each class has 200 to 400 images and most images has medium resolutions. Table 3 shows results of classification experiments using 100 images per class for training and the remaining for testing. The results indicate our method underperformed on this data set. In contrast, spatial pyramid match method which utilized global scene features performed well [11]. We conjecture that the worse performance is because our method is not suitable for *scene* classification. Most images in this dataset usually do not have dominant objects at the center and scenes usually changes from image to image, it is difficult to apply our proposed approach to learn the pairwise relationship between two images from this kind of dataset using only local keypoint matching information.

	SVMχ_2	**SIM-O**	**SIM**	**Pyramid** [11]
Acc	51.3±0.5	53.8±0.5	58.4±0.6	81.4±0.5

Table 3: Results on 15 scene categories dataset.

6. CONCLUSIONS

In this paper, we have presented a semi-supervised learning approach to image matching tasks with several novelties in the feature representation. Our approach uses only local descriptors, without any other image features such as shapes and edges. By formulating the matching task as a semi-supervised problem, a large amount of unlabeled images can be leveraged with a small amount of labeled image pairs to boost matching performance. We believe other visual information such as color, shape and geometric context can further improve performance by tailoring these features for the specific applications. One of underlying problems associated with SIM is that each image is actually characterized by salient objects (where rich local descriptors can be found and matched). This can be good or bad. For object oriented image retrieval, it is good because users want to find images including the same objects, but SIM may not perform well if the user wants to find images which including uniform textured objects or regions such as a blue sky or beach. In the future, we will study how to cope with multiple objects in single image and change of object poses between images.

7. ACKNOWLEDGMENTS

This work was partially supported by gift grants from General Motors and Microsoft.

8. REFERENCES

[1] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. COLT, 1998.

[2] W.J. Christmas, J. Kittler, and M. Petrou. Matching of Road Segments Using Probabilistic Relaxation: reducing the computational requirements. Imaging and Vision for control and guidance of aerospace vehicles, 2004.

[3] R. Ewerth and B. Freisleben. Semi-supervised learning for semantic video retrieval. CIVR, 2007.

[4] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. PAMI, 2006.

[5] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. CVPR, 2003.

[6] L. Grady and G. Funka-Lea. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. ECCV Workshop, 2004.

[7] K. Grauman and T. Darrell. Unsupervised learning of categories from sets of partially matching image features. CVPR, 2006.

[8] T. Hertz, N. Shental, A. Bar-Hillel, and D. Weinshall. Enhancing image and video retrieval: Learning via equivalence constraints. CVPR, 2003.

[9] A. Kostin, J. Kittler, and W.J. Christmas. Object recognition by symmetrised graph matching using relaxation labelling with an inhibitory mechanism. Pattern Recognition Letters, 2005.

[10] H. Kuck, P. Carbonetto, and N. de Freitas. A constrained semi-supervised learning approach to data association. ECCV, 2004.

[11] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. CVPR, 2006.

[12] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[13] C. L. Mallows. A note on asymptotic joint normality. *Annals of Math. Stat.*, 1972.

[14] K. Mikolajczyk and C. Schmid. An affine invariant interest point. ECCV, 2002.

[15] C.W. Ngo, W.L. Zhao, and Y.G. Jiang. Fast tracking of near-duplicate keyframes in broadcast domain with transitivity propagation. ACM Multimedia, 2006.

[16] A. Rebai, A. Joly, and N. Boujemaa. Interpretability based interest points detection. CIVR, 2007.

[17] J. Ros, C. Laurent, and G. Lefebvre. A cascade of unsupervised and supervised neural networks for natural image classification. CIVR, 2006.

[18] C. Rosenberg, M. Hebert, and H. Schneiderman. Semi-supervised selftraining of object detection models. WACV, 2005.

[19] F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets. ECCV, 2002.

[20] N. Sebe, T. Gevers, J. van de Weijer, and S. Dijkstra. Corner detectors for affine invariant salient regions: Is color important?. CIVR, 2006.

[21] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T.Freeman. Discovering object categories in image collections, 2005. MIT-CSAIL-TR-2005-012.

[22] J. Stottinger, J. Amores, N. Sebe, A. Hanbury, N. Boujemaa, and T. Gevers. Object categorisation with color interest points. CIVR, Demo, 2007.

[23] Q. Tian, J. Yu, Q. Xue, and N. Sebe. A new analysis of the value of unlabeled data in semi-supervised learning for image retrieval. ICME, 2004.

[24] J. Weng, N. Ahuja, and T. S. Huang. Matching two perspective views. PAMI, 1992.

[25] W. Wu and J. Yang. Smartlabel: An object labeling tool using iterated harmonic energy minimization. ACM Multimedia, 2006.

[26] H. Zhang, A. Berg, M. Maire, and J. Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. CVPR, 2006.

[27] W. Zhao, Y.G. Jiang, and C.W. Ngo. Keyframe retrieval by keypoints: Can point to point matching help?. CIVR, 2006.

[28] Z.H. Zhou, K.J. Chen, and Y. Jiang. Exploiting unlabeled data in content-based image retrieval. ECML, 2004.

[29] X.J. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. ICML, 2003.

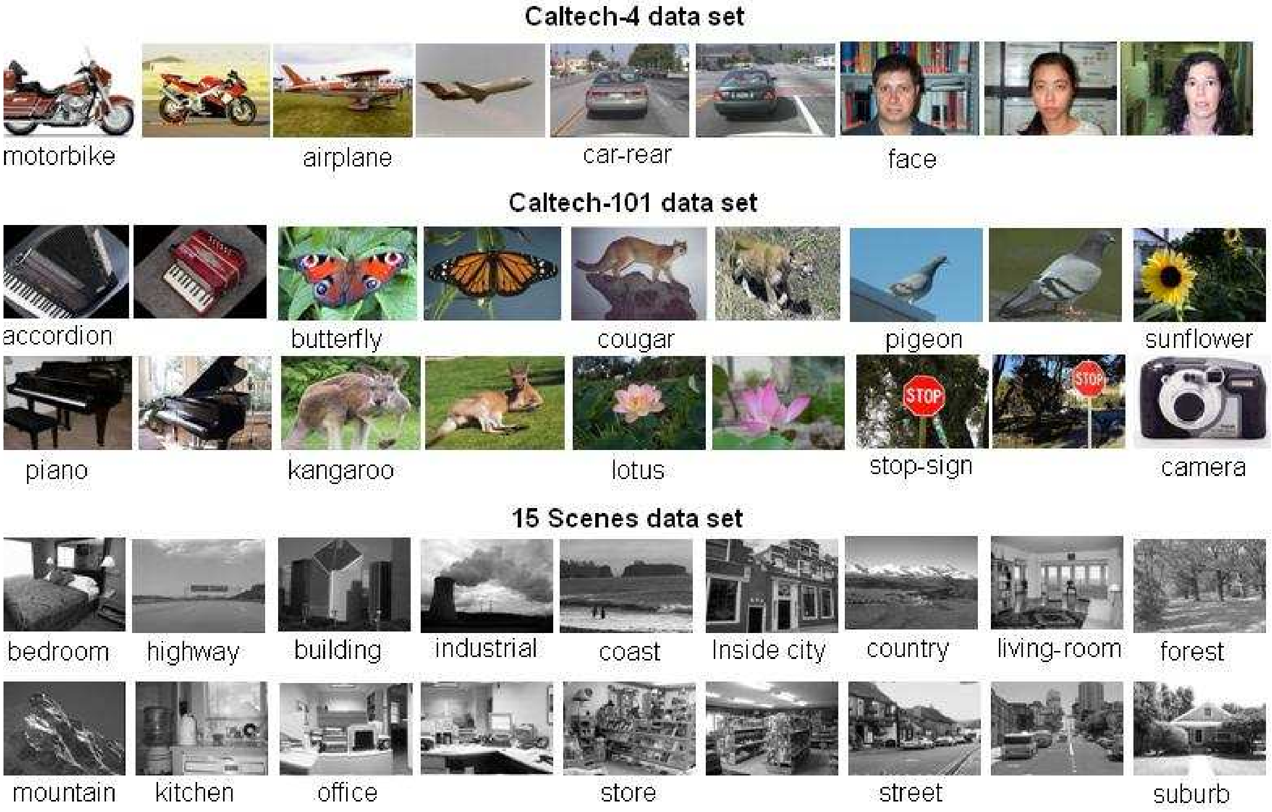
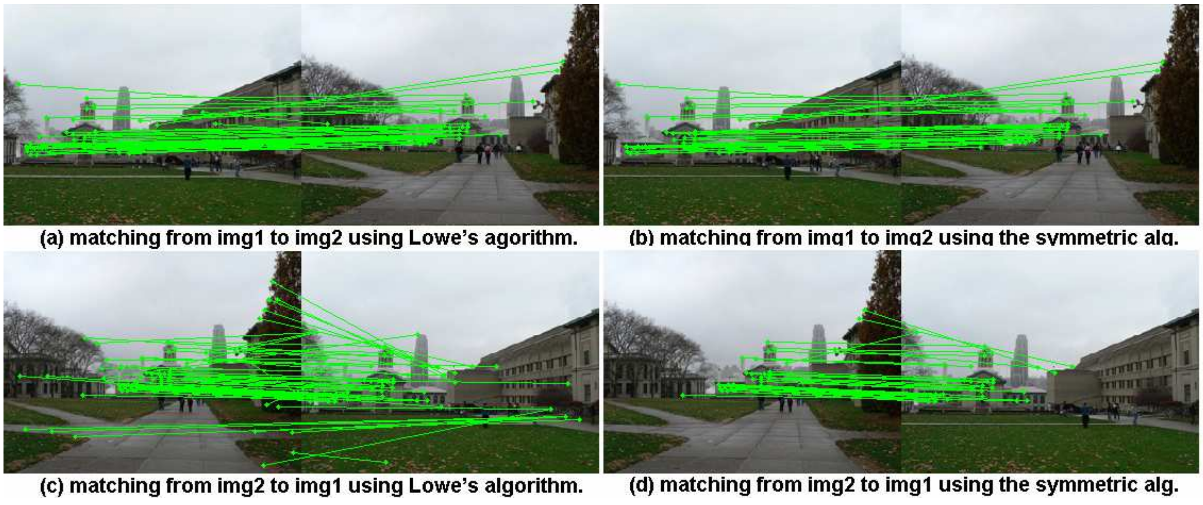


Figure 8: Example images from 3 diverse datasets which are used in our experiments.


(a) matching from img1 to img2 using Lowe's agorithm.

(b) matching from img1 to img2 using the symmetric alg.

(c) matching from img2 to img1 using Lowe's algorithm.

(d) matching from img2 to img1 using the symmetric alg.

Figure 9: Comparing Lowe's matching criterion [12] and our sysmetrical criterion. Notice the asymmetrical matchings by comparing (a) and (c). In constrast, each selected local keypoint in both images in (b) can be found in (d). ***Notice that, from Img1 to Img2 both methods find the same matchings, (a) and (b), but our method only keeps symmetrical ones which can be found from matching from Img2 to Img1 in (d).***