# One-Shot Multi-Set Non-rigid Feature-Spatial Matching

Marwan Torki　　　　　　Ahmed Elgammal
Department of Computer Science
Rutgers University, New Brunswick, NJ, USA
{mtorki,elgammal}@cs.rutgers.edu

## Abstract

*We introduce a novel framework for nonrigid feature matching among multiple sets in a way that takes into consideration both the feature descriptor and the features spatial arrangement. We learn an embedded representation that combines both the descriptor similarity and the spatial arrangement in a unified Euclidean embedding space. This unified embedding is reached by minimizing an objective function that has two sources of weights; the feature spatial arrangement and the feature descriptor similarity scores across the different sets. The solution can be obtained directly by solving one Eigen-value problem that is linear in the number of features. Therefore, the framework is very efficient and can scale up to handle a large number of features. Experimental evaluation is done using different sets showing outstanding results compared to the state of the art; up to 100% accuracy is achieved in the case of the well known 'Hotel' sequence.*

## 1. Introduction

Finding correspondences between features in different images plays an important role in many computer vision tasks. Several robust and optimal approaches have been developed for finding consistent matches for rigid objects by exploiting a prior geometric constraint [23]. The problem becomes more challenging in a general setting, *e.g.*, matching features on an articulated object, deformable object, or matching between two instances (or a model to an instance) of the same object class for recognition and localization. For such problems, many researchers recently tend to use high-dimensional descriptors encoding the local appearance, (e.g. SIFT features [13]). Using such highly discriminative features makes it possible to solve for correspondences without much structure information or avoid solving for correspondences all together, which is quite popular trend in object categorization [19]. This is also motivated by avoiding the high complexity of solving for spatially consistent matches.

The problem we address in this paper is how to find matches between *multiple* sets of features where both the feature descriptor similarity and the spatial arrangement of the features need to be enforced. However, the spatial arrangement of the features needs to be encoded and enforced in a relaxed manner to be able to deal with non-rigidity, articulation, deformation, and within class variation.

The problem of matching appearance features between two images in a spatially consistent way has been addressed recently (e.g. [11, 5, 3, 22]). Typically this problem is formulated as an attributed graph matching problem where graph nodes represent the feature descriptors and edges represent the spatial relations between features. Enforcing consistency between the matches led researchers to formulate this problem as a quadratic assignment problem where a linear term is used for node compatibility and a quadratic term is used for edge compatibility. This yields an NP-hard problem [3]. Even though some efficient solutions (e.g. linear complexity in the problem description length) have been proposed for such a problem [5] the problem description itself remains quadratic, since consistency has to be modeled between every pair of edges in the two graphs. This puts a huge limitation on the applicability of such approaches to handle large number of features[1].

Besides this scalability limitation, most of the state of the art algorithms for matching can only match two sets of points. They do not generalize to match multiple sets of features.

In this paper, we introduce a framework for feature matching among multiple sets of features in one shot, where both the feature similarity in the descriptor space, as well as the local spatial geometry are enforced. *This formulation brings three contributions to the problem:*
*1) Graph Matching through Embedding:* We formulate the problem of consistent matching as an embedding problem where the goal is to embed all the features in a Euclidean

[1] For example, for matching $n$ features in two images, an edge compatibility matrix of size $n^2 \times n^2$, i.e., $O(n^4)$, needs to be computed and manipulated to encode the edge compatibility constraints. Obviously this is prohibitively complex and does not scale to handle a large number of features.

embedding space where the locations of the features in that space reflect both the descriptor similarity and the spatial arrangement. This is achieved through minimizing an objective function enforcing both the feature similarity and the spatial arrangement. Such embedding space acts as a new unified feature space (encoding both the descriptor and spatial constraints) where the matching can be easily solved. The framework is illustrated in Fig 1.

*2) Matching Multiple sets in one shot:* The proposed framework directly generalizes to matching multiple sets of features in one shot through solving one Eigen-value problem. Consistent matching of multiple sets of features is a fundamental problem, for which very few solutions have been proposed.

*3) Scalability:* An interesting point in this formulation is that the spatial arrangement for each set is only encoded within that set itself, *i.e.*, in a graph matching context no compatibility needs to be computed between the edges (no quadratic terms or higher order terms), yet we can enforce spatial consistency. Therefore the proposed approach is scalable and can deal with hundreds and thousands of features. Minimizing the objective function in the proposed framework can be done by solving an Eigen-value problem *which size is linear in the number of features in all images.*

Extensive evaluation on several standard datasets shows that the proposed approach gives better or comparable results to the state of the art algorithms [11, 5, 3, 22] that uses quadratic assignment. In fact, we achieve 100% correct matching on a standard benchmark with our multiset setting. The experiment results also show that the proposed approach can find consistent matching under wide range of variability including: 3D-motion, viewpoint change, rotation, zooming, blurring, articulation and nonrigid deformation.

## 2. Related Work

There is a huge volume of literature on matching features given a class of geometric transformation between two images or a model to an image. However, more related to our work, are recent papers on matching highly discriminative local appearance features under relaxed geometric constraints [11, 5, 3, 22, 10] which are more geared towards dealing with nonrigidity and within-class variability.

There is a huge literature on formulating correspondence finding as a graph-matching problem. We refer the reader to [3] for an excellent survey on this subject. Matching two sets of features can be formulated as a bipartite graph matching in the descriptor space, *e.g.* [1], and the matches can be computed using combinatorial optimization, *e.g.* the Hungarian algorithm [17]. Alternatively, spectral decomposition of the cost matrix can yield an approximate relaxed solution, *e.g.* [18, 6]. Alternatively, matching can be formulated as a graph isomorphism problem between two weighted or unweighted graphs to enforce edge compatibility, *e.g.* [24, 21, 25]. The intuition behind such approaches is that the spectrum of a graph is invariant under node permutation and, hence, two isomorphic graphs should have the same spectrum, the converse does not hold. Several approaches formulated matching as a quadratic assignment problem and introduced efficient ways to solve it, *e.g.* [9, 2, 5, 11, 22]. Such formulation enforces edgewise consistency on the matching. We discussed the limitations of such approaches in Section 1. Even, higher order consistency terms have been introduces [7]. In [3] an approach was introduced to learn the compatibility functions from examples and was found that linear assignment with such a learning scheme outperforms quadratic assignment solutions such as [5]. Our experiments show that we can reach similar or better results without resorting to higher order compatibility terms.

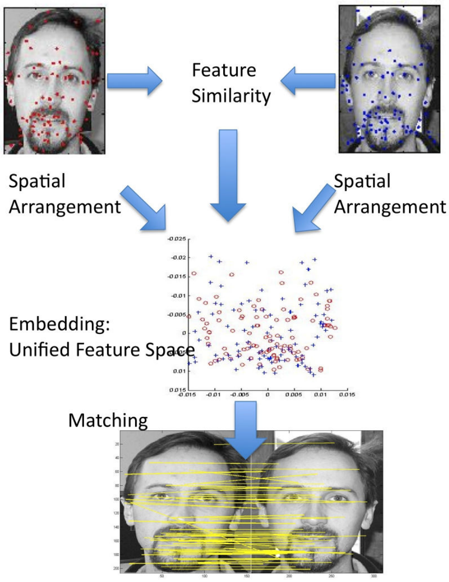

Figure 1. Motivative Example on two faces

Matching multiple sets in image sequences can be addressed by forward tracking a set of features [20]. There are very few papers that addressed solving for multiset correspondences in a fundamental way, *e.g.* [16, 4]. In [4] a simulated annealing-like approach was introduced to find correspondences between multiple point sets and was used to obtain shape average. However, the solution deals only with point features (no appearance). Multiset correspondences can also be found through clustering in the descriptor space. Such solution is popular in object recognition to obtain a visual codebook [19]. However such solution ignores the spatial consistency.

## 3. Feature Embedding Framework

### 3.1. Problem Statement

We are given $K$ sets of feature points, $X^1, X^2, \cdots X^K$ in $K$ images where $X^k = \{(x_1^k, f_1^k), \cdots, (x_{N_k}^k, f_{N_k}^k)\}$ Each feature point $(x_i^k, f_i^k)$ is defined by its spatial location in its image plane $x_i^k \in \mathbb{R}^2$ and its feature descriptor $f_i^k \in \mathbb{R}^D$, where $D$ is the dimensionality of the feature descriptor space[2]. For example, the feature descriptor can be a SIFT, HOG, etc. Notice that the number of features in each set might be different. We use $N_k$ to denote the number of feature points in the $k$-th point set. Let $N$ be the total number of points in all data sets, i.e., $N = \sum_{k=1}^K N_k$.

There are two kinds of information that need to be preserved:
1) Feature similarity: feature descriptors in general represents the appearance in a way that is assumed to be invariant to viewing conditions.
2) spatial structure of each data set represents the shape or the arrangement of the features.

To achieve a model that preserves these two constraints we introduce two data kernels based on the affinities in the spatial and descriptor domains separately. The spatial affinity (structure) in each image can be represented by a weight matrix $\mathbf{S}^k$ where $\mathbf{S}_{ij}^k = K_s(x_i^k, x_j^k)$ and $K_s(\cdot, \cdot)$ is a spatial kernel local to the $k$-th image that measures the spatial proximity. The feature affinity between image $p$ and $q$ can be represented by the weight matrix $\mathbf{U}^{pq}$ where $\mathbf{U}_{ij}^{pq} = K_f(f_i^p, f_j^q)$ and $K_f(\cdot, \cdot)$ is a feature kernel that measures the similarity in the descriptor domain between the $i$-th feature in image $p$ and the $j$-th feature in image $q$. Here we describe the framework given any spatial and feature weights in general and in Section 5 we will give specific details on the kernels we use.

We are looking for an embedding for the all feature points into a common embedding space. Let $y_i^k \in \mathbb{R}^d$ denotes the embedding coordinate of point $x_i^k$, where $d$ is the dimensionality of the embedding space, *i.e.*, we are seeking a set of embedded point coordinates $Y^k = \{y_1^k, \cdots, y_{N_k}^k\}$ for each input feature set $X^k$. The embedding should satisfy the following two constraints

- The feature points from different point sets with high feature similarity should become close to each other in the resulting embedding as long as they do not violate the spatial structure.
- The spatial structure of each point set should be preserved in the embedding space and should not be affected by false feature matches (*i.e.*, should not be pulled away by false matches).

Let us jump ahead and assume an embedding can be achieved satisfying the aforementioned spatial structure and the feature similarity constraints. Such an embedding space represents a new "Feature" space that encodes both the features' descriptor and the spatial structure information. Given such an embedding, the matching problem between two sets reduces to solving a Bipartite graph matching between the two sets of embedded coordinates $Y^p$ and $Y^q$ where the weights between the two sets are mainly based on the Euclidean distances in the embedding space. Matching multiple sets reduces to a clustering problem in the Euclidean embedding space.

Embedding all the input points in such a way will result in a consistent set of matches, which means the pairs of matches will obey some common transformation between the two point sets. Therefore, there is no need to explicitly add pairwise consistency constraints as done in quadratic matching approaches [2, 5, 11, 22].

### 3.2. Objective Function

Given the above stated goals, we reach the following objective function on the embedded points $Y$, which need to be minimized

$$\Phi(Y) = \sum_k \sum_{i,j} \|y_i^k - y_j^k\|^2 \mathbf{S}_{ij}^k + \sum_{p,q} \sum_{i,j} \|y_i^p - y_j^q\|^2 \mathbf{U}_{ij}^{pq}, \tag{1}$$

where $k$, $p$ and $q = 1, \cdots, K$ and $p \neq q$. The objective function is intuitive; the first term preserves the spatial arrangement within each set, since it tries to keep the embedding coordinates $y_i^k$ and $y_j^k$ of any two points $x_i^k$ and $x_j^k$ in a given point set close to each other based on their spatial kernel weight $\mathbf{S}_{ij}^k$. The second term of the objective function tries to bring close the embedded points $y_i^p$ and $y_j^q$ if their feature similarity kernel $\mathbf{U}_{ij}^{pq}$ is high.

This objective function can be rewritten using one set of weights defined on the whole set of input points as:

$$\Phi(Y) = \sum_{p,q} \sum_{i,j} \|y_i^p - y_j^q\|^2 \mathbf{A}_{ij}^{pq}, \tag{2}$$

where the matrix $\mathbf{A}$ is defined as

$$\mathbf{A}_{ij}^{pq} = \begin{cases} \mathbf{S}_{ij}^k & p = q = k \\ \mathbf{U}_{ij}^{pq} & p \neq q \end{cases} \tag{3}$$

where $\mathbf{A}^{pq}$ is the p-q block of $\mathbf{A}$.

The matrix $\mathbf{A}$ is an $N \times N$ weight matrix with $K \times K$ blocks where the $p - q$ block is of size $N_p \times N_q$. The $k$-th diagonal block is the spatial structure kernel $\mathbf{S}^k$ for the $k$-th set. The off-diagonal $p - q$ block is the descriptor similarity kernels $\mathbf{U}^{pq}$. The matrix $\mathbf{A}$ is symmetric by definition since diagonal blocks are symmetric and since $\mathbf{U}^{pq} = \mathbf{U}^{qp^T}$. The matrix $\mathbf{A}$ can be interpreted as a weight matrix between points on a large point set where all the input points

[2] Throughout this paper, we will use superscripts to indicate a dateset (image) index and subscripts to indicate point index within the set, i.e., $\boldsymbol{x}_i^k$ denotes point $i$ in the $k$-th set.

are involved in this point set. Points from a given data set are linked be weights representing their spatial structure $\mathbf{S}^k$; while nodes across different data sets are linked by suitable weights representing their feature similarity kernel $\mathbf{U}^{pq}$.

We can see that the objective function Eq. 2 reduces to the problem of Laplacian embedding [15] of the point set defined by the weight matrix $\mathbf{A}$. Therefore, the objective function reduces to

$$\mathbf{Y}^* = \arg \min_{\mathbf{Y}^T\mathbf{D}\mathbf{Y}=\mathbf{I}} tr(\mathbf{Y}^T\mathbf{L}\mathbf{Y}), \quad (4)$$

where $\mathbf{L}$ is the Laplacian of the matrix $\mathbf{A}$, i.e., $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{D}$ is the diagonal matrix defined as $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$ . The $N \times d$ matrix $\mathbf{Y}$ is the stacking of the desired embedding coordinates such that,

$$\mathbf{Y} = \left[y_1^1, \ldots, y_{N_1}^1, y_1^2, \ldots, y_{N_2}^2, \ldots y_1^K, \ldots, y_{N_K}^K\right]^T$$

The constraint $\mathbf{Y}^T\mathbf{D}\mathbf{Y} = \mathbf{I}$ removes the arbitrary scaling. Minimizing this objective function is a straight forward generalized eigenvector problem: $\mathbf{L}y = \lambda\mathbf{D}y$. The optimal solution can be obtained by the bottom $d$ nonzero eigenvectors. The required $N$ embedding points $Y$ are stacked in the $d$ vectors in such a way that the embedding of the points of the first point set will be the first $N_1$ rows followed by the $N_2$ points of the second point set, and so on.

The objective function in Eq 2 is general. We can easily see that matching algorithms that use only spatial constraints are a special case by replacing the off-diagonal blocks in the affinity matrix $\mathbf{A}$ by a constant block. On the other hand, matching algorithms that use the feature similarity constraints only is a special case by replacing the diagonal blocks in the affinity matrix $\mathbf{A}$ by an identity block.

Notice that the size of the matrix $\mathbf{A}$ is linear in the number of input points, i.e., for the case of matching two sets, $\mathbf{A}$ is an $(N_1 + N_2) \times (N_1 + N_2)$ matrix. In contrast, other approaches that enforces pairwise consistency [2, 5, 11, 22] use a consistency matrix that is quadratic in size $N_1N_2 \times N_1N_2$. Such quadratic order hinders the scalability of such matching techniques.

# 4. Feature Matching

## 4.1. Matching Settings

The embedding achieved through minimizing the objective function Eq 2 represents a Euclidean "Feature" space encoding both the descriptors' similarity and the local spatial structures. Solving for matching will be a straight forward task in such space. We present three settings in which our framework can be used depending on the application.

**Pairwise Matching (PW):** Given two sets of features, the matching reduces to solving a bipartite graph matching problems between two sets of embedding coordinates. We give details about how to obtain the matching in Sec 4.2.

**Multiset Pairwise Matching (MP):** If we have multiple sets of features and we would like to find pairwise matching between each pair of sets, then embedding all the features in all the sets will give a global unified feature space. Pairwise matches between any two sets can also be solved as a bipartite graph matching where the weights are defined in the embedding coordinates. In this case, the global solution is expected to enhance the pairwise solution. This is shown in the experiment in Sec 6.2. We give details about how to obtain the matching in Sec 4.2.

**Multiset Clustering (MC):** If we have multiple sets of feature points the unified embedding should bring correspondent features from different sets to be close to each other. In that sense, clustering can be used to in the embedding space to obtain matching features. In this paper we applied k-means clustering in the embedding coordinate to find the feature groups. Other clustering techniques can be used. The problem can also be formulated as a Multi-partite graph matching in the embedding space.

In Sec. 6.2 we show the results obtained by applying these three settings on the well known 'Hotel' sequnece.

## 4.2. Matching Criterion

The embedding coordinates achieved by solving the objective function 1 guarantees that the Euclidean distances between the embedded points reflect both the spatial and descriptor constraints. Therefore, the matching problem reduces to solving a bipartite matching problem in the embedding space. This can be solved by many approaches such as the Hungarian algorithm [17] and others. However, in particular we used the Scott and Longuet-Higgins (SLH) algorithm [18] as matching criterion in the embedding space. The conditions required for the Scott and Longuet-Higgins matching are satisfied by the embedding since all the points are lying on the same plane and there are no large rotation. We compute an $N_1 \times N_2$ Ecuildian distance based weight matrix $\mathbf{W}$ in the embedding space using a Gaussian kernel and then we compute an orthonormal matrix $\mathbf{P}^*$ in a way similar to Eq. 5. We decide a match if the element $\mathbf{P}^*_{ij}$ is maximum in its row and its column. In addition we add the condition that the second largest element in its row and its column is far by threshold ratio as done in [6].

The main reason we chose the SLH algorithm over the Hungarian algorithm as a matching criterion is its ability to reject false matches. The Hungarian algorithm finds a matching for each feature even though that match might not be good, which is not a desired characteristic.

## 5. Feature and Spatial Affinities

### 5.1. Spatial Structure Weights

The spatial structure weight matrix $S^k$ should reflect the spatial arrangement of the features in each set $k$. In general, it is desired that the spatial weight kernel be invariant to geometric transformations. However, this is not always achievable. In this paper we used two different kinds of spatial weights: 1) Euclidean-based weights: the weights are based on the Euclidean distances between features defined in each image coordinate system. Such weights are invariant to translation and rotations. 2) Affine invariant-based weights: any three non-colinear points in an image defines basis for an affine invariant coordinate system. We use three matches between two images to obtain an affine parameterization of all the other features in each images. Alternatively, subspace invariance [26] can be used to obtain an affine invariant coordinate system.

Several kernels can be used to obtain the spatial weights based on either the Euclidean coordinates or the affine coordinates including the Gaussian kernel defined as $S^k_{ij} = e^{-\|x^k_i - x^k_j\|^2/2\sigma^2}$, and the Double exponential kernel defined as $S^k_{ij} = e^{-\|x^k_i - x^k_j\|/\sigma}$ Our evaluation shows that both the Gaussian and double exponential gives comparable results.

### 5.2. Feature Weights

The feature weight matrix $\mathbf{U}^{pq}$ should reflect the feature to feature similarity in the descriptor space between the $p$-th and $q$-th sets. A seemingly obvious choice is the widely used affinity based on a Gaussian kernel on the squared Euclidean distance in the feature space, i.e., $\mathbf{G}^{pq}_{ij} = e^{-\|f^p_i - f^q_j\|^2/2\sigma^2}$ given a scale $\sigma$. However, such choice is not suitable for the purpose of the objective function in Eq 1. This is mainly because such weights do not satisfy the exclusion principle. One feature from an image can be similar to many features in the second image. The objective function will try to bring all these similar features close to each other in the embedding space which might violate the spatial arrangement.

The feature weights should reflect the feature similarity and, in the same time, should satisfy the exclusion principle. On the other hand, we should avoid making any hard decision on the matching from the feature similarity alone, i.e., a zero-one permutation matrix is not a suitable feature weight matrix. In other words, the feature weights should be soft correspondence weights. To achieve this we solve for the feature weights in a way similar to the Scott and Longuet-Higgins algorithm [18].

Given the feature affinity $\mathbf{G}$ between features in sets $p$ and $q$, we need to solve for a permutation matrix $\mathbf{C}$ that permutes the rows of $\mathbf{G}$ in order to maximize its trace, i.e.,

$$\psi(\mathbf{C}) = tr(\mathbf{C}^T\mathbf{G})$$

The permutation matrix constraint can be relaxed into an orthonormal matrix constraint on the matrix $\mathbf{C}$. Therefore, the goal is to find optimal an orthonormal matrix $\mathbf{C}^*$ such that

$$\mathbf{C}^* = \arg \max_{s.t.\mathbf{C}^T\mathbf{C}=\mathbf{I}} tr(\mathbf{C}^T\mathbf{G}) \quad (5)$$

It was shown in [18] that the optimal solution for 5 is

$$\mathbf{C}^* = \mathbf{UEV}^T$$

where the SVD decomposition of $\mathbf{G} = \mathbf{USV}^T$ and $\mathbf{E}$ is obtained by replacing the singular values on the diagonal of $\mathbf{S}$ by ones. The orthonormal matrix $\mathbf{C}^*$ are used as the feature weights $\mathbf{U}^{pq} = \mathbf{U}^{qp^T}$ after setting the negative values to 0.

## 6. Experiments

In this section we show both quantitative and qualitative results on different data set. Despite that our focus is on non-rigid matching, we also show results on rigid matches for quantitative and comparative evaluation [3].

### 6.1. Non-Rigid Matching

Fig. 2 shows some matching results on nonrigid motions. We used sequences from the KTH dataset [4]. Fig. 2-top shows the results of our pairwise matching (**PW** setting) using SIFT features on four frames of a walking motion, *i.e.*, 6 pairs. Our approach boosted the matches obtained to double of the original SIFT matches. Fig. 2-bottom shows the result of the multiset setting (**MC**) applied on 13 frames of a half cycle of hand waving. Due to the low resolution in the sequence, a small number of features are detected (around 25 features per frame). Enforcing the global matching with the spatial constraints boosted the number of matches to from 44 to 73 and correct matches can be found on the moving parts for all the 13 frames.

Fig. 3 shows sample matches on motorbike and airplane images from Caltech101 [12]. In each case we used eight images and used the Multiset Pairwaise (**MP**) to match all pairs. In these experiments we used affine kernels and Geometric Blur [2] features.

### 6.2. Comparative Evaluation: 3D Motion (Wide Baseline Matching)

**Goal:** This experiment aims at evaluating our proposed framework compared to the state of the art reported results including linear and quadratic assignment based approaches [5, 3, 22, 25, 10, 8] .
**Data:** We use the CMU ‘Hotel’ sequence with the same manual labeling of 30 landmark points employed in [3].

[3] To the best of our knowledge there is no available non-rigid dataset with ground-truth matches.

[4] http://www.nada.kth.se/cvap/actions/

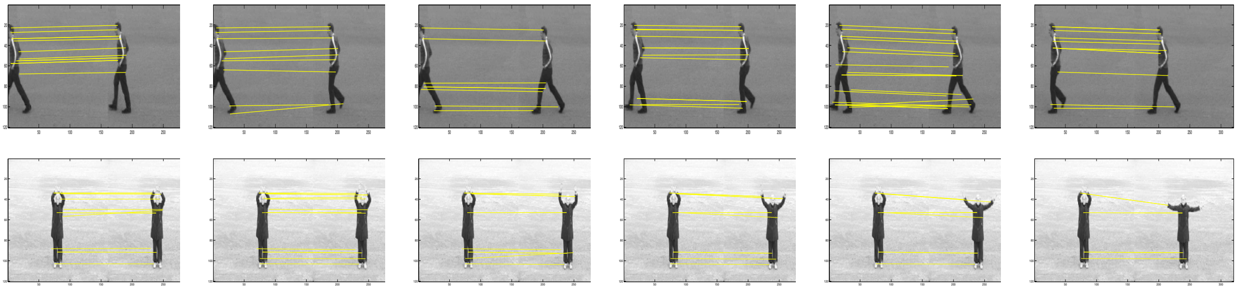

Figure 2. Top: Results on non rigid walking sequence (matched pairwise). Bottom: Sample results on hand waving sequence matched on a 13 frames in one shot (multiset). Shown is the first image matches with the consequtive odd frames in the 13 frames

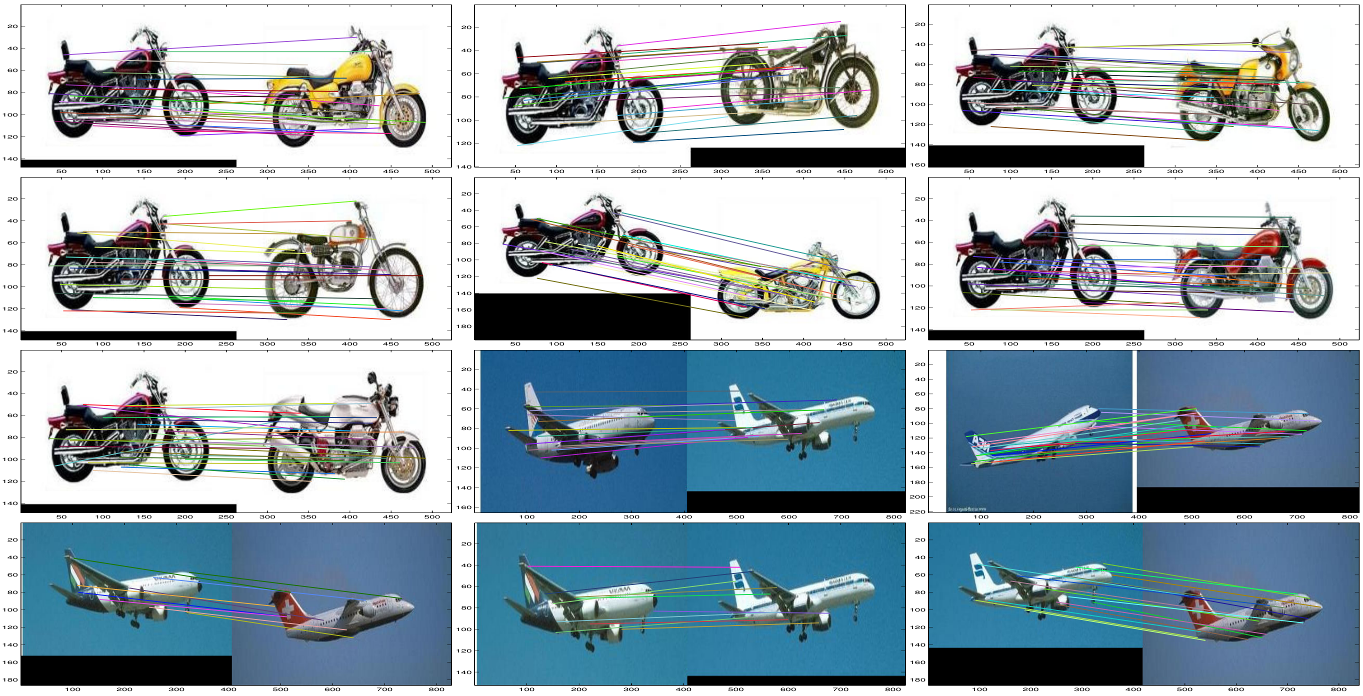

Figure 3. Sample results on Caltech 101 images. Best seen in color. Sample pairs are shown here, all pairs are shown in the supplementary materials )

This dataset has been used in [3, 22] to compare the performance of graph matching methods. The sequence contains 101 frames that shows a 3D motion of the 'Hotel' object. The experiment is done using the same setting as [3, 22]: 15 frames are sampled (every 7 frames), that gives 105 pairs of images to match.

**Competitive Approaches:** In all cases we used the Shape context [1] as the feature descriptor (except for KPCA). We compared the following: 1)The KPCA matching [25] is an example of an algorithm that only uses the spatial structure. 2) Descriptor-only linear assignment: we used the Hungarian algorithm applied to the shape context descriptor. In this case only feature similarity is used. We used the histogram distances as our metric as it was introduced in [1]. 3) Our approaches: The three settings described in Sec 4.1: Pairwise (**PW**), Multiset pairwise (**MPW**) and Multiset with clustering (**MC**). We used a Euclidean double exponential kernel to encode the spatial structure, and Gaussian kernel on the *same* shape context descriptor for descriptor similarity. 4) Dual Decomposition approach proposed in [22]. This is a quadratic assignment approach that uses an iterative solution. 5) Results reported in [22], which are state of the art algorithms using quadratic optimizations. That includes [5] a spectral relaxation of the graduated assignment, [10, 8] and max-product belief propagation on a quadratic pseudo-boolean optimization [22]. 6) Results reported in [3] after learning on another sequence (CMU 'House' sequence) using both quadratic and linear assignment with learning.

**Evaluation:** Evaluation is based on the mismatch ratio and the complexity of the problem. Table 1 shows that our basic

| Algorithm | Error Rate | Problem complexity |
|---|---|---|
| KPCA [25] | 35.5% | Linear |
| Linear Assign. W/SC [17] | 11.81% | Linear |
| Our Approach PW | **9.24%** | Linear |
| Our Approach MPW | **4.44%** | Linear |
| Our Approach MC | **0.0%** | Linear |
| SMAC [5] | 15.97% | Quadratic |
| Fusion [10] | 13.05% | Quadratic |
| COMPOSE [8] | 4.51% | Quadratic |
| Belief Propagation [22] | 0.06% | Quadratic |
| Dual Decomposition [22] | 0.19% | Quadratic |
| Learning(LA) [3] | 12-17% | Linear |
| Learning(GA) [3] | 10-14% | Quadratic |

Table 1. State of the art results on the 'Hotel' Sequence

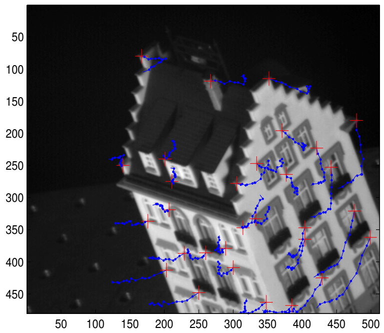

Figure 4. Matches obtained in 15 frames of the 'Hotel' sequence using one-shot multiset matching

**PW** outperforms all approaches that use linear complexity and outperforms some of the state of the art quadratic algorithms, e.g., [5, 10]. Using our multiset **MPW** and **MC** we reach 95.56% and 100% accuracy, which is not reached by any of the competing algorithms. It is very important to notice that the size of our affinity matrix $A$ in the case of the multiset of 15 frames is just $450 \times 450$ and for the case of the pairwise matching is $60 \times 60$, where the size for one edge compatibility matrix for any of the quadratic assignment approaches is $900 \times 900$. Table 1 shows the complexity of the problem and the mismatch ratio. Fig 4 shows the matches obtained from all the 15 frames using our multiset approach.

### 6.3. Robustness: INRIA datasets

**Data:** In this experiment we use the INRIA datasets, which has been used by [14] for comparing descriptors. This dataset contains seven subsets that covers several effects such as viewpoint change, zooming, rotation, blurring and lighting change. Each of the seven datasets has a ground truth *Homography* matrix computed between the first image in each set and the other images in same dataset. Overall there are 36 matching problems given their ground truth.

| Dataset(Effect) | SIFT<br>Matching [13] | SVD on SIFT<br>Matching [6] | Our<br>Approach | Our Affine<br>Approach | $1^{st}$ Image<br>Feature Count |
|---|---|---|---|---|---|
| Graf (ViewPoint) | 47 | 54 | 66 | **67** | 464 |
| Boat (Zoom&Rotation) | 99 | 87 | **108** | **108** | 467 |
| Bark (Zoom&Rotation) | 49 | 47 | **55** | **55** | 392 |
| Bricks (ViewPoint) | 46 | 44 | 58 | **59** | 310 |
| Trees (Blurring) | 146 | 153 | 186 | **191** | 642 |
| Cars (Lighting) | 60 | 17 | 65 | **70** | 134 |
| Bikes (Blurring) | 227 | 229 | **239** | 237 | 400 |

Table 2. Average number of correct matches for each dataset from INRIA datasets

**Goal:** We use the INRIA data set to evaluate the robustness of the pairwise matching version of our framework to the various imaging effect in a dataset with ground truth. We also evaluate the behavior of the matching under strong affine transformation using both the Euclidean and the affine invariant kernels. This set demonstrates the scalability of our approach to handle a very large number of feature points ( from 130 to 1250 SIFT features per image). That shows the value of our approach compared to the quadratic assignment approaches, which typically can only handle a number of features limited to around 100. We use the ground truth *Homography* matrices just for evaluating the resulting matches, since our approach does not assuming any geometric transformation prior.

**Competitive Approaches:** in this experiment we compare 1) The basic SIFT matches [13] as a baseline. 2) SVD-SIFT [6]: This approach uses SVD decomposition on a Gaussian proximity matrix in the SIFT descriptor space. 3) Our Pairwise matching approach with both a Euclidean Gaussian spatial kernel and an affine invariant kernel. In all cases we are using the same set of SIFT descriptors.

**Results:** Table 2 shows that for all the datasets, our approach with either kernels gives the highest number of correct matches. The last column gives the number of features in the first image for each dataset. This result shows that enforcing the spatial consistency improves the descriptor matches. Fig. 5 shows the number of matches as a function of the viewpoint change or the blurring[5]. The results show that the Euclidean kernel gives comparable results to the affine invariant kernel even under a very large viewpoint change. We selected the scale for the spatial kernel as a constant-multiple of the maximum distance between feature points in each image. In general, we found that selecting a scale large enough for the Euclidean kernels would give results comparable to affine invariant kernels, this is consistent with what was stated in [18]. Matching results between images can be seen in the supplemental materials.

## 7. Conclusion and Future Work

This paper shows that we can enforce spatial consistency for matching high-dimensional local appearance features in an efficient and scalable way. The embedding formulation introduced encodes both the inter sets feature similarity and

[5] more plots can be seen in the supplementary materials

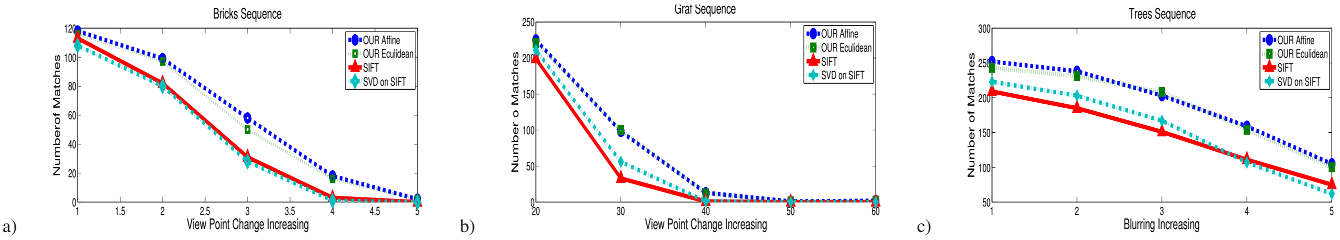

Figure 5. Number of matches affected by Different effects. a,b) Increasing view point Change(Bricks and Graf), c) Increasing Blurring (Trees)

the intra sets spacial structure in a unified space. This combination of constraints is shown to be enough to achieve consistent matching. Since spacial structure is only measured within each set, there is no need to for quadratic edge consistency constraints. Therefore, the approach is linear and can scale to deal with large numbers of features. Pairwise matching based on the proposed framework was shown to give comparable and even better results than quadratic assignment approaches. The framework can be directly applied to match multiple sets, which was shown to outperform all the reported state of the art results. The approach can match multiple sets by solving a single eigenvalue problem on matrix which size is linear in the number of features. The experiments also shows that the approach always has a very low false matching rates, *i.e.*, it is biased towards getting high certainty matches. Further theoretical and empirical studies are needed to understand how to control the matching to be biased towards enforcing rigidity vs. enforcing descriptor similarity.

**Acknowledgments:** This research is partially funded by NSF CAREER award IIS-0546372.

## References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *TPAMI*, 2002. 2, 6

[2] A. C. Berg. *Shape Matching and Object Recognition*. PhD thesis, University of California, Berkeley, 2005. 2, 3, 4, 5

[3] T. S. Caetano, J. J. McAuley, L. Cheng, Q. V. Le, and A. J. Smola. Learning graph matching. *TPAMI*, 2009. 1, 2, 5, 6, 7

[4] H. Chui, A. Rangarajan, J. Zhang, and C. M. Leonard. Unsupervised learning of an atlas from unlabeled point-sets. *TPAMI*, 2004. 2

[5] T. Cour, P. Srinivasan, and J. Shi. Balanced graph matching. *NIPS*, 2006. 1, 2, 3, 4, 5, 6, 7

[6] E. Delponte, F. Isgrò, F. Odone, and A. Verri. Svd-matching using sift features. *Graph. Models*, 2006. 2, 4, 7

[7] O. Duchenne, F. Bach, I. S. Kweon, and J. Ponce. A tensor-based algorithm for high-order graph matching. *CVPR*, 2009. 2

[8] J. Duchi, D. Tarlow, G. Elidan, and D. Koller. Using combinatorial optimization within max-product belief propagation. *NIPS*, 2007. 5, 6, 7

[9] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *TPAMI*, 1996. 2

[10] V. Lempitsky, C. Rother, and A. Blake. Logcut: Efficient graph cut optimization for markov random fields. *ICCV*, 2007. 2, 5, 6, 7

[11] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. *ICCV*, 2005. 1, 2, 3, 4

[12] F. Li, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *CVIU*, 106(1):59–70, April 2007. 5

[13] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 1, 7

[14] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *TPAMI*, 2005. 7

[15] P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 2003. 4

[16] R. Oliveira, R. Ferreira, and J. P. Costeira. Optimal multi-frame correspondence with assignment tensors. *ECCV*, 2006. 2

[17] C. Papadimitriou and K. Stieglitz. *Combinatorial Optimization Algorithms and Complexity*. Prentice Hall, 1982. 2, 4, 7

[18] G. Scott and H. Longuett-Higgins. An algorithm for associating the features of two images. *The Royal Society of London*, 1991. 2, 4, 5, 7

[19] T. Serre, L. Wolf, and T. Poggio. Object recognition with features inspired by visual cortex. In *CVPR 2005*, 2005. 1, 2

[20] K. Shafique and M. Shah. A noniterative greedy algorithm for multiframe point correspondence. *TPAMI*, 2005. 2

[21] L. Shapiro and J. Brady. Feature-based correspondence: an eigenvector approach. *Image and Vision Computing*, 1992. 2

[22] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization. *ECCV*, 2008. 1, 2, 3, 4, 5, 6, 7

[23] S. Ullman. Aligning pictorial descriptions: An approach to object recognition. *Cognition*, 1989. 1

[24] S. Umeyama. An eigen decomposition approach to weighted graph matching problems. *TPAMI*, 1988. 2

[25] H. Wang and E. R. Hancock. Correspondence matching using kernel principal components analysis and label consistency constraints. *PR*, 2006. 2, 5, 6, 7

[26] Z. Wang and H. Xiao. Dimension-free afne shape matching through subspace invariance. *CVPR*, 2009. 5