Image Clustering with Metric, Local Linear Structure, and Affine Symmetry

Jongwoo Lim[1], Jeffrey Ho[2], Ming-Hsuan Yang[3],
Kuang-chih Lee[1], and David Kriegman[2]

[1] University of Illinois at Urbana-Champaign, Urbana, IL 61801
[2] University of California at San Diego, La Jolla, CA 92093
[3] Honda Research Institute, Mountain View, CA 94041

Abstract. This paper addresses the problem of clustering images of objects seen from different viewpoints. That is, given an unlabelled set of images of n objects, we seek an unsupervised algorithm that can group the images into n disjoint subsets such that each subset only contains images of a single object. We formulate this clustering problem under a very broad geometric framework. The theme is the interplay between the geometry of appearance manifolds and the symmetry of the 2D affine group. Specifically, we identify three important notions for image clustering: the L^2 distance metric of the image space, the local linear structure of the appearance manifolds, and the action of the 2D affine group in the image space. Based on these notions, we propose a new image clustering algorithm. In a broad outline, the algorithm uses the metric to determine a neighborhood structure in the image space for each input image. Using local linear structure, comparisons (affinities) between images are computed only among the neighbors. These local comparisons are agglomerated into an affinity matrix, and a spectral clustering algorithm is used to yield the final clustering result. The technical part of the algorithm is to make all of these compatible with the action of the 2D affine group. Using human face images and images from the COIL database, we demonstrate experimentally that our algorithm is effective in clustering images (according to ojbect identity) where there is a large range of pose variation.

1 Introduction

Given a collection of images, one may wish to group or cluster the images according to many different attributes of the images and their content. For instance, one may wish to cluster them based on some notion of human categories or taxonomies of objects. Or one might wish to cluster based on scene content (e.g., beach, agricultural, or urban scenes). Or perhaps one might wish to cluster all images into groups with the same lighting or with the same pose (this might only be relevant for images from a specific class such as faces [1]). In this paper, we consider the problem of clustering images according to the identity of the 3D objects, but where the observer's viewpoint has varied between images.

T. Pajdla and J. Matas (Eds.): ECCV 2004, LNCS 3021, pp. 456–468, 2004.

Clearly, this type of image clustering problem requires understanding how the images of an object vary under different viewing conditions, and so the goal of the clustering algorithm is to detect some consistent patterns among the images. A traditional computer vision approach to solve this problem would most likely include some kind of image feature extraction, e.g., texture, shape, filter bank outputs, etc. [2,3]. The underlying assumption is that some global or local image properties of a 3D object exist over a wide range of viewing conditions. The drawback of such an approach is that it is usually difficult to extract these features reliably and consistently. Appearance-based approaches e.g. [4,5] offer a different kind of strategy for tackling the clustering problem. For this type of algorithm, image feature extraction no longer plays a significant role. Instead, it is the geometric relations among images in the image space that is the focus of attention. The geometric concept that is central to appearance-based methods is the idea of an appearance manifold introduced in [6].

Our goal is to identify certain crucial geometric elements, such as the appearance manifold, that are central to the image clustering problem and to formulate a new clustering algorithm accordingly. Specifically, the two main contributions of this paper are:

1. We formulate the image clustering problem under a very general geometric framework. Using this framework, we provide a clear geometric interpretation of our algorithm and comparisons between our work and previous image clustering algorithms.
2. Motivated by geometric considerations, we propose a new image clustering algorithm.

We have tested our algorithm on two types of image data: images in the Columbia COIL database and images of human faces. Images of the 3D objects in the COIL database have more variation in surface texture and shape. Therefore, local image features can be extracted more reliably from these images [2]. For images of human faces, the variations in texture and shape are much more limited, and any clustering algorithm employing feature extractions is not expected to do well. We will show that our algorithm is capable of producing good clustering results for both types of image data.

2 Clustering Algorithm

In this section, we detail our image clustering algorithm. Schematically, our algorithm is similar to other clustering algorithms proposed previously, e.g., [7, 4]. That is, we define affinity measures between all pairs of images. These affinity measures are represented in a symmetric $n \times n$ matrix $A = (a_{ij})$, i.e., the affinity matrix and a straightforward application of any standard spectral clustering method [8,9] then yields our clustering result. The machinery employed to solve the clustering problem, i.e. spectral clustering, has been studied quite intensively in combinatorial graph theory [10], and it is of no concern to us here. Instead, our focus is on 1) explaining the geometric motivation behind our algorithm and 2) the definition of the affinity a_{ij}.

First, we define our image clustering problem. The input of the problem is a collection of unlabelled images $\{I_1, \cdots, I_n\}$ and the number of clusters N. We assume that all images have the same number of pixels s, and by rasterizing the images, we obtain a collection of corresponding sample points $\{x_1, \cdots, x_n\}$ in $\mathbb{R}^s$. Our algorithm outputs a cluster assignment for these images $\rho : \{I_1, \cdots, I_n\} \to \{1, \cdots, N\}$. Two images I_i and I_j belong to the same cluster if and only if $\rho(I_i) = \rho(I_j)$. A cluster, in our definition, consists of only images of one object. We further assume that the images of a cluster are acquired at different view points but under the same ambient illumination condition.

The problem so formulated is extremely general and without any further information, there is almost no visible structure to base the algorithm on. One obvious structure one can utilize is the ambient distance metric of the image space. The usual L^2 metric or its derivatives (affine-invariant L^2 distance or weighted L^2 distance) are such examples. By considering images as points in $\mathbb{R}^s$, we are naturally led to the notion of appearance manifolds [6]. Accordingly, the input images imply the existence of N sub-manifolds of $\mathbb{R}^s$, $\{M_1, \cdots, M_N\}$ such that two points x_i, x_j belong to the same cluster if and only if $x_i, x_j \in M_k$ for some $1 \leq k \leq N$, with each M_i denoting the appearance manifold of an object. Implicit in the concept of appearance manifolds is the idea of local linearity. That is, if $x_1, \cdots, x_l$ are points belonging to the same cluster and if they are sufficiently close according to the distance metric, then each point x_i can be well-approximated linearly by its neighbors : $x_k \approx \sum_{j \neq k} a_j x_j$ for some real numbers a_j.

Metric and local linearity are two very general geometric notions and they do not pertain only to image clustering problems. It is the action of the 2D affine group G [1] that characterizes our problem as an image clustering problem rather than a general data clustering problem. If $\{x_1, \cdots, x_n\}$ were data of a different sort, e.g. data from a meteological or high energy physics experiment, there will not be an explicit action of G. It is precisely because the 2D nature of the images and the way we rasterize the image to form points in $\mathbb{R}^s$, we can explicitly calculate the action of G given a sample point x. In particular, each appearance manifold M_i is invariant under G, i,e, if $x \in M_i$ then $\gamma(x) \in M_i$ for each $\gamma \in G$. In this sense, the clustering problem acquires a symmetry played by the 2D affine group[2]. In summary, we have identified three important elements to the image clustering problem. First, there is the ambient L^2 (and its derivatives) metric of the image space. Second, each cluster has local linear structure. The metric and local linearity are the only two geometric structures we can utilize in designing the algorithm. The third element is the affine symmetry of the problem. Our challenge is to design a clustering algorithm that takes into account these three elements. In a very general outline, what is needed is to design metric and local linear structure that are both invariant under the affine group G *and* to seek an interesting and effective coupling between the metric and

[1] Henceforth, except at a few places, G will invariably denote the 2D affine group.

[2] Strictly speaking, the symmetry group will depend on what type of imaging model is used for the problem. In general, it will be a subgroup of G rather than G itself.

linear structure, which are two rather disparate geometric notions. Surprisingly, using only these three very general structures, we can formulate a clustering algorithm which will be demonstrated to be effective for a variety of image clustering problems. Our algorithm is compact and purely computational. Many standard vision techniques, such as local feature extractions and PCA, will not make their appearances in our algorithm.

The clustering problem we studied here is considerably more difficult than the illumination clustering studied in [11]. The main difference between their case and ours is a difference between global and local. In the illumination case, the linear structure, the illumination cone, is a global structure and it can be exploited directly in designing the clustering algorithm. In our case, the linear structure is only a local structure and unlike the cone which admits a compact and precise description via its generators, our local linear structure is more difficult to quantify. Therefore, the exploitation of the local linear structure in our algorithm is more subtle than in the illumination case.

2.1 Metric Structure

Since the input images are considered as a collection of points in $\mathbb{R}^s$, the usual L^2-distance metric and its derivatives offer the simplest affinity measures between a pair of data points. However, since the clusters form manifolds in $\mathbb{R}^s$, they are not expected to localize in some region of $\mathbb{R}^s$ independently of other clusters. This observation can be supported by the fact that the Euclidean distance between two face images of different identities acquired at the same pose is almost always smaller than the Euclidean distance of two images of the same identity but acquired at different poses [12,13]. Two analogous situations in 3D are depicted in Figure 1. They clearly demonstrate that if the metric information is used for defining affinity, then "medium" and "long-distance" comparisons are usually erroneous.

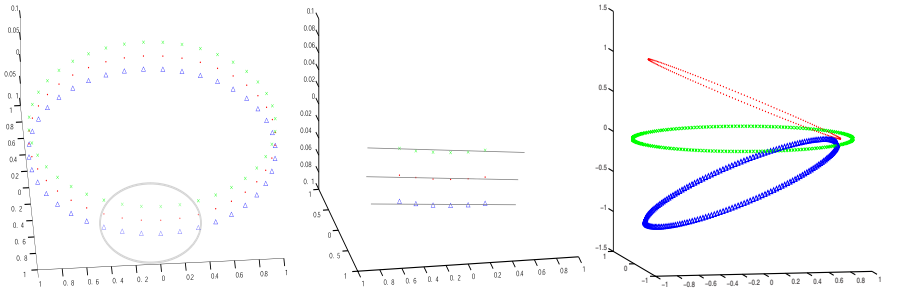

Fig. 1. Left(A) Three parallel circles. The points on each circle are uniformly sampled and the distance between adjacent circles is slightly smaller than the distance between two neighboring points on the same circle. **Center(B)** A "magnified" view of a neighborhood. **Right(C)** The top and bottom circles are rotated by $\pm 30°$.

However, Figure 1(B) suggests one good way of using the metric is not to use it directly for comparison. Instead, we can use the metric to pick data points for which the comparisons will be made. In particular, for each point x, the metric defines a neighborhood and in this neighborhood, non-metrical information can be exploited to do the comparison (i.e., defining affinity). In this way, the metric defines a collection of local clustering problems, and the affinities computed in these local settings will then be put into the global affinity matrix to provide a final clustering result.

2.2 Local Linear Structure (LLS)

Figure 1 shows two examples which are unlikely to be clustered correctly using the metric information along. Figure 1(A) is a good example. The data collection contains points sampled uniformly from three circles in $\mathbb{R}^3$. The distance between adjacent circles are slightly smaller than the distance between two neighboring points on the same circle. To the best of our effort, we can not correctly cluster the data into three circles using only metric information. The point of course is that the manifold structure of the circles must be taken into consideration. One possible way to use the manifold structure is to compute the "tangent space" at each sample point using Principal Component Analysis in a neighborhood of the sample point, as in [4]. This approach can correctly cluster Figure 1(A) but unlikely[3] to cluster Figure 1(C) correctly. This is mainly because the local linear estimate using PCA becomes unstable in the region when the circles come into close contact with each other.

Instead of working with tangents, we shift our focus slightly to consider the secant approximation of a sample point by its neighbors, see Figure 2(A). For a smooth 2D curve, each point x can be approximated well by a point on the secant chord formed by two of its sufficiently close neighbors y_1, y_2: $x \approx a_1 y_1 + a_2 y_2$ with a_1, a_2 non-negative and $a_1 + a_2 = 1$. This can be generalized immediately to higher dimension: for a point x and its neighbors, $\{y_1, \cdots, y_K\}$, we can try to compute a set of non-negative coefficients ω_i which is the solution to the following optimization problem:

$$\min \left\| x - \sum_{i=1}^{K} \omega_i y_i \right\|_{L^2}^2 \tag{1}$$

with the constraint that $\sum_{i=1}^{K} \omega_i = 1$. Assuming $\{y_1, \cdots, y_K\}$ are linearly independent[4], then the coefficients ω_i are unique. Figure 2(B) illustrates that the magnitude of the coefficients ω_i can be used as an affinity measure locally to detect the presence of any linear structure. That is, a large magnitude of ω_i indicates the possibility that y_i and x share a common local linear structure.

[3] To the best of our effort!

[4] In the image space $\mathbb{R}^s$, this is almost always true since $K \ll s$.

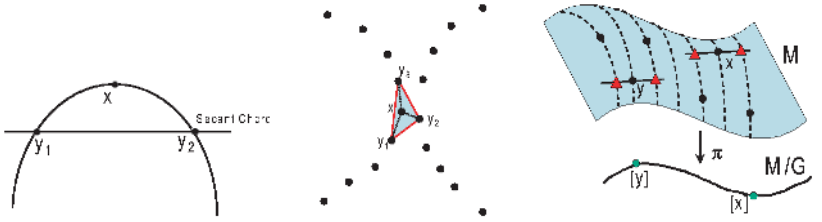

Fig. 2. Left(A) The secant chord approximation of a point on a smooth curve by its neighbors. **Center(B)** Two semi-circles. There are three possible secant chord approximations of x by the three sides of the shaded triangle. **Right(C)** The shaded surface denotes the appearance manifold M and the dashed lines are the orbits of the affine group G. The projection map π sends each point x in M to the corresponding point $[x] \in M/G$. The solid circles denote the sample points. In order to construct the local linear structure in M/G, we have to move the sample points along the orbits to produce "virtual" samples, denoted by the triangles.

Applying the idea we have outlined so far, a simple data clustering algorithm can be designed[5], and we can cluster all the examples above correctly. On the other hand, to our best effort, we can't find a simple and straightforward algorithm, based on the more traditional clustering techniques such as the K-means and connected component analysis, etc., that can successfully cluster all of these examples.

2.3 Affine Symmetry and Quotient Spaces

As we mentioned earlier, the presence of the 2D affine group distinguishes the image clustering problem from the general data clustering problem. The task now is to put both the metric and local linear structure into an affine invariant setting (as best as we can). Affine invariant L^2 metric and many of its variants have been studied before in the literature [5,14,15], etc. Our effort is to propose a method for defining local linear structures that are affine invariant; in particular, we want to reformulate Equation 1 in an affine-invariant way.

We will explain this with the mathematical notion of a quotient space [16]. In general, when there is a group G acting on a manifold M, one can associate this action with an abstract topological space M/G, the quotient space. Loosely speaking, the space M/G parameterizes the orbits of the group action. See Figure 2(C). The important point is that any quantity defined in M that is invariant under the G-action can be naturally defined as a derived quantity in the space M/G. For instance, if we have a G-invariant metric on M, this metric then in turn defines a metric on M/G.

Specializing to our clustering problem, the manifold M is the union of the appearance manifolds, $\{M_1, \cdots, M_N\}$, and the group G is the 2D affine group.

[5] Compute ω_i for each sample point using its K-nearest neighbors provided by the metric. Form a symmetric affinity matrix using ω_i and apply the spectral clustering.

We have the natural projection map $\pi : M \to M/G$ which takes each point x of M to the point $[x] \in M/G$ which parameterizes the orbit containing x. The manifolds $\{M_1, \cdots, M_N\}$ now descend down to M/G to form $\{\tilde{M}_1, \cdots, \tilde{M}_N\}$. By speaking of affine invariant local linear structure, we are speaking of the local linear structures of these "manifolds" [6], $\{\tilde{M}_1, \cdots, \tilde{M}_N\}$.

To compute the local linear structures of $\{\tilde{M}_1, \cdots, \tilde{M}_N\}$, we can mimic the standard slice construction for quotient spaces [16]. The idea is that for each point $[x]$ of M/G, we can compute its local linear structure by lifting the computation to a sample point $x \in M$ such that $\pi(x) = [x]$. At each such point x, we take a "slice" of the group action, i.e., a linear subspace centered at x that is orthogonal to the G-action through x and we analyze the local linear structures on the slice. See Figure 2.(C). For each sample point x we find a slice S. We project all other sample points down to S using G, i.e. for a sample point y, we find a $\gamma \in G$ such that $\gamma(y) \in S$. Note that such γ may not exist for every y but we only need a few such ys to characterize a neighborhood of x. Let $\{y'_1, \cdots, y'_s\}$ be the projected points on S. We use the L^2 metric in S to select the right neighbors of x, say, $\{y'_1, \cdots, y'_K\}$ and use them in defining the local linear structure at $[x]$ via Equation 1.

In the actual implementation, we modify the slice construction outlined above. Instead of actually computing the subspace S, we determine the K neighbors $\{y'_1, \cdots, y'_K\}$ by using the "one-sided distance" [14]. For each input sample y, the "one-sided distance" is defined as

$$d_G(x, y) = \min_{\gamma \in G} \Big\{ \min \{ \|x - \gamma(y)\|^2_{L^2}, \|y - \gamma(x)\|^2_{L^2} \} \Big\}.$$

Although $d_G(x, y)$ is not a metric, it still allows us to define the K-nearest neighbors of x. The K neighbors $\{y'_1, \cdots, y'_K\}$ of x above are just $\{\gamma_1(y_1), \cdots, \gamma_K(y_K)\}$ with each γ_i minimizes the one-sided distance between x and y_i.

3 Related Work

In this section, we compare our algorithm with some of the well-known image clustering algorithms in the literature. Needless to say, the 2D affine group has a long history in the computer vision literature. In particular, intensive effort has been focused on studying (quasi-) affine invariant metric such as the tangent distance e.g. [15,17]. For image clustering, affine invariant metric has made its appearance in the work of Fitzgibbon and Zisserman [5,14]. Most of the effort in these two papers has been focused on designing an affine invariant metric that will be effective for clustering. In the language of the quotient space, they are

[6] Demonstrating the quotient space is actually some "nice" geometric object is generally a very delicate mathematical problem [16]. It is not our intention here to rigorously define the space M/G. Our goal is to use the idea of quotient space to explain the motivation of the algorithm and in the next section, to compare our algorithm with other previous algorithms.

doing clustering on M/G using metric information alone. Our algorithm also uses the metric information in M/G but it also explicitly tries to cluster " manifolds" in M/G. Although good clustering results can be obtained by considering metric alone, we believe that by incorporating both the metric and local linearity, it offers 1) a more effective clustering algorithm and 2) a more complete geometric description of the clustering algorithm.

Another well-known image clustering algorithm that explicitly uses the concept of the appearance manifold is [4]. However, there are two major differences between our work and theirs. First, the affine symmetry is absent in [4]. One of the main themes of this paper is that the action of the 2D affine group is of central importance in formulating any image clustering problem. Second, there is an important difference between our concept of local linearity and theirs. In [4], the concept of local linearity is embodied in the idea of tangent space of the appearance manifold; therefore, PCA is used to estimate local linear subspaces. In contrast, our concept of local linearity is on how best the "neighbors" can linearly approximate a given sample point, and it is formulated through Equation 1. This concept of local linearity also allows non-geometric interpretation in terms of image comparisons using parts of objects as in [18]; however, it is not clear if there is a non-geometric interpretation of the tangent spaces used in [4].

[2,3] are two other interesting and related papers on image clustering. Their approaches and ours are fundamentally different in that our algorithm is completely image-based while their algorithms focus on extracting salient image features and incorporating more sophisticated machine learning techniques for clustering. However, comparisons between their experimental results and ours will be made in the next section.

4 Experiments

In this section, we report our experimental results. Our image clustering algorithm, as detailed in Figure 3, has been implemented in MATLAB. Two different types of image data were used to test the algorithm, images of 3D objects and images of human faces. Substantial variations in appearances are observed in all image datasets. The main difference between these two types of datasets is the variation in surface texture. For the former type, the surface texture varies greatly and local image features (such as corners) can be more reliably extracted. Human faces, on the other hand, have much limited variation in surface texture and local image features become less useful. Traditionally, these two different types of image data were attacked separately using feature-based methods (e.g. [2]) and appearance-based methods (e.g. [1]), respectively. However, the results below show that our algorithm is capable of obtaining good clustering results for both types of images.

Except for the affine-invariant metric $d_G(x, y)$, the implementation is straightforward and it follows closely the steps outlined in Figure 3. Given two images, I_1, I_2, $d_G(I_1, I_2)$ is computed as follows. First, we define a Gaussian distribution p on 2D affine group centered at the identity. Since we only consider

1. **Inputs**
   A collection of unlabelled images $\{I_1, \cdots, I_n\}$. Considered the images as data points $\{x_1, \cdots, x_n\}$ in the image space $\mathbb{R}^s$, and the number of clusters N.
2. **Use Metric to Choose Neighbors**
   For each data point x, compute a set of K nearest neighbors using the distance measure $d_G(x, y)$ defined above.
3. **Use local linear structure**
   For each x and its K-neighbors $\{x_1, \cdots, x_K\}$ determined in the previous step, let $\{y_1, \cdots, y_K\}$ be the points in $\mathbb{R}^s$ such that $y_i = \gamma(x_i)$ for some $\gamma \in G$ and y_i minimizes the distances between x and all points on the orbit of G through x_i. Using y_i's to linearly approximate x by determining a collection of K non-negative real numbers ω_i that minimizes the objective function
   $$\left\| x - \sum_{i=1}^{K} \omega_i y_i \right\|_{L^2}^2, \qquad \text{with the constraint that } \sum_{i=1}^{K} w_i = 1$$
4. **Use ω_i as the affinity measure**
   Define an affinity measure d_Ω between two data points x_i and x_j: $d_\Omega(x_i, x_j) = \min(1/\omega_{ij}, 1/\omega_{ji})$ where ω_{ij} is the coefficient computed in the previous step for x_i. If x_j is not among the K-neighbors of x_i, ω_{ij} is set to 0. Apply the spectral clustering algorithm (e.g., [8]) using this affinity to yield the final clustering result.

Fig. 3. The clustering algorithm.

small affine corrections, p can be expressed in a local coordinates system centered at the identity by expressing each (small) affine transformation in terms of the usual six parameters (a 2x2 matrix plus translation). Using these six parameters, p is a Gaussian distribution with diagonal covariance matrix. Next, we determine an affine transformation γ such that it minimizes the function

$$E(\gamma) \;=\; \min\left\{\, d_{L^2}(\gamma(I_1), I_2),\, d_{L^2}(I_1, \gamma(I_2)) \,\right\}$$

where d_{L^2} is the usual L^2 distance metric between two images. γ can be found using gradient descent [19]. $d_G(I_1, I_2)$ is then defined as the sum $E(\gamma) - \log p(\gamma)$. The reason for incorporating the Gaussian $p(\gamma)$ is to penalize "over-corrections" by large affine transformations [14].

4.1 Datasets

In this subsection, we fix the notations for various image datasets we used in the experiments and give brief descriptions of the datasets. For images of 3D objects, we use the COIL datasets from Columbia, which are popular datasets for validating object recognition algorithms. There are two COIL datasets, COIL20 and COIL100. They contain 20 and 100 objects, respectively. For both datasets, the images of each object were taken 5 degrees apart as the object is rotated on a turntable and each object has 72 images. Since this sampling is quite dense, we "sub-sampled" the image collections to make clustering problem more interesting. We let COIL20.2 denote the collection of images obtained from COIL20

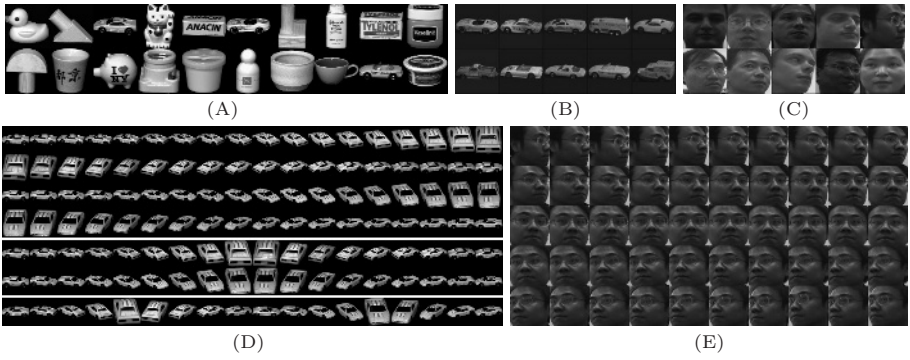

Fig. 4. (A): Representative images of objects in COIL20 **(B)**: The ten vehicles in VEH10.2 **(C)**: The ten individuals in FACE10 **(D)**: Sampling frequency. First 4 rows are images of one object from COIL20, next 2 rows from COIL20.2, and last row from COIL20.4 **(E)**: Pose variation in FACE10.

by sub-sampling it with a factor of 2. So COIL20.2 contains the same number of objects as the original COIL20 but with half as many images per object. Similarly, COIL20.4 denotes the collection obtained from COIL20 by sub-sampling it with a factor of 4 and so on. From COIL100.2, we placed all vehicle images in this collection together to form a new dataset, VEH10.2. The images of these vehicles have similar appearances and therefore, they offer a challenging dataset to test our algorithm. For images of human faces, we collected video sequences of ten individuals to form ten image sequences with each sequence containing 50 images. Pose variation in this collection is quite large and because of the differences in individual motion, the image sequences do not have uniform variation in pose. This dataset will be denoted by FACE10.

4.2 Results

The experimental results are reported in Table 1. As is clear from Table 1, our algorithm produces good clustering results for all datasets except COIL100.4. The algorithm's performance on COIL100 is not surprising considering that there are 100 objects in COIL100.4 and the images are rather sparsely sampled (every 20 degrees). Error rates are calculated as the ratio of the number of misclustered images over the number of images[7]. The error rates are shown together with the parameter K which defines the size of the local neighborhoods. We also mention that there are clustering results on COIL20 database reported in [2]. We can not translate their definition of errors into ours. However, they do report non-zero error rate while our clustering algorithm achieves a perfect clustering result for the COIL20 dataset.

[7] For each cluster emerged from the clustering result, we try to match it with the known clusters (ground-truth). Once the one-to-one map between the new clusters and known clusters is computed, the error ratio can be calculated accordingly. For instance, a random assignment of a collection of N clusters of equal size will produce an error rate of $\frac{N-1}{N}$ according to our definition.

Table 1. Clustering results of our algorithm

	FACE10	COIL20	COIL20.2	COIL20.4	VEH10.2	COIL100.2	COIL100.4
Error	0.00%	0.00%	5.14%	19.44%	11.11%	20.69%	34.89%
K	10	8	8	8	3	6	10

Table 2. Comparison with other clustering algorithms

Algorithms	Datasets			
	COIL20.2	COIL20.4	FACE10	VEH10.2
Our algorithm	5.14%	19.44%	0.00%	11.11%
Affine+K-NN+Spectral	7.36%	21.11%	13.00%	27.50%
Affine+Spectral	10.14%	25.83%	22.00%*	40.00%
Euclidean+Spectral	35.14%	33.06%	25.60%*	61.67%
Euclidean+K-means	39.58%	48.06%	46.00%	74.44%

* Spectral clustering results indicate the results may not be robust

4.3 Comparison with Other Clustering Algorithms

Table 2 lists the result of comparing (on four different datasets) our algorithm with some standard off-the-shelf algorithms. First, two standard clustering algorithms, K-means and spectral clustering algorithm [8] with the usual L^2-distance metric, are compared with our results. It clearly demonstrates that direct L^2 comparisons without affine-invariance are not sufficient at all. Next, we incorporate affine-invariance but without using local comparisons (Affine+Spectral). This is the "one-sided" distance measure [14] and again, it is still not able to produce good clustering results. Next, we show that by incorporating local linear structure in the algorithm, it does indeed enhance the performance of the clustering algorithm. Note that in our framework, once a neighborhood structure has been determined, we exploit the local linear structure to cluster points in the neighborhood. To show that this is indeed effective and necessary, we replace this step of our algorithm with direct metric comparisons. That is, we are computing local affinities based purely on the "one-sided" distance measure (Affine+K-NN+Spectral). We expect that our algorithm will be an improvement over this method because of our use of non-metrical information, and the results do indeed corroborate our claim.

Finally, in Figure 5, we illustrate several results of our local linear estimates, i.e., the ω_i. Although the K-nearest neighbors of an image generally contain

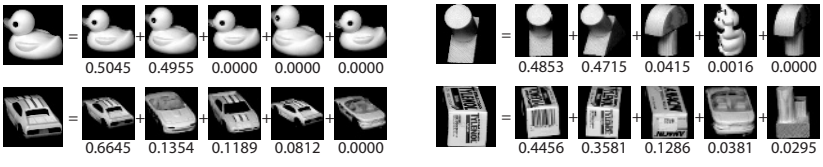

Fig. 5. Images, their neighbors and the local linear structure, ω_i's.

images of other objects, in each case, ω_i correctly pick out the right images to form strong affinities.

5 Concluding Remarks

In this paper, we have proposed an image clustering algorithm, and we have demonstrated with a number of experiments that our algorithm is indeed effective for clustering images of 3D objects undergoing large pose variation. One obvious limitation of our algorithm is that we do not explicitly model the illumination effect. However, [11] has demonstrated that it is possible to cluster images with illumination variation using global linear structures. How best to incorporate our local structure and the global one in [11] into an effective image clustering algorithm that can deal with both lighting and pose variations will be a challenging and interesting research direction for the future.

Acknowledgements. This work was funded under NSF CCR 00-86094, NSF CCR 00-86094, the U.C. MICRO program, and the Honda Research Institute.

References

1. Li, S., Lv, X., Zhang, H.: View-based clustering of object appearances based on independent subspace analysis. In: Proceedings of IEEE International Conference on Computer Vision. (2001) 295–300
2. Saux, B.L., Boujemaa, N.: Unsupervised robust clustering for image database categorization. In: International Conference on Pattern Recognition. Volume 1. (2002) 259–262
3. Frigui, H., Boujemaa, N., Lim, S.: Unsupervised clustering and feature discrimination with application to image database categorization. In: Joint 9th IFSA World Congress and 20th NAFIPS Conference. (2001)
4. Basri, R., Roth, D., Jacobs, D.: Clustering appearances of 3D objects. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (1998) 414–420
5. Fitzgibbon, A.W., Zisserman, A.: On affine invariant clustering and automatic cast listing in movies. In Heyden, A., Sparr, G., Nielsen, M., Johansen, P., eds.: Proceedings of the Seventh European Conference on Computer Vision. LNCS 2353, Springer-Verlag (2002) 304–320
6. Murase, H., Nayar, S.K.: Visual learning and recognition of 3-D objects from appearance. In: International Journal of Computer Vision. Volume 14. (1995) 5–24
7. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 888–905
8. Ng, A., Jordan, M., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In Ditterich, T., Becker, S., Ghahramani, Z., eds.: Advances in Neural Information Processing Systems 15, MIT Press (2002) 849–856
9. Weiss, Y.: Segmentation using eigenvectors: A unifying view. In: Proceedings of IEEE International Conference on Computer Vision. Volume 2. (1999) 975–982

10. Chung, F.R.K.: Spectral Graph Theory. American Mathematical Society (1997)
11. Ho, J., Yang, M.H., Lim, J., Lee, K.C., Kriegman, D.: Clustering appearances of objects under varying illumination conditions. In: IEEE Conf. on Computer Vision and Pattern Recognition. Volume 1. (2003) 11–18
12. Graham, D.B., Allinson, N.M.: Norm^2-based face recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Volume 1. (1999) 586–591
13. Raytchev, B., Murase, H.: Unsupervised face recognition from image sequences based on clustering with attraction and repulsion. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Volume 2. (2001) 25–30
14. Fitzgibbon, A.W., Zisserman, A.: Joint manifold distance: a new approach to appearance based clustering. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Volume 1. (2003) 26–33
15. Simard, P., Cun, Y.L., Denker, J., Victorri, B.: Transformation invariance in pattern recognition - tangent distance and tangent propagation. In: Neural Networks. Volume 1524. (1998) 239–274
16. Mumford, D., Kirwan, F., Fogarty, J.: Geometric Invariant Thoery. Springer-Verlag (1994)
17. Frey, B., Jojic, N.: Fast, large-scale transformation-invariant clustering. In: Advances in Neural Information Processing Systems 14. (2001) 721–727
18. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. Nature **401** (1999) 781–791
19. Hager, G.D., Belhumeur, P.N.: Efficient region tracking with parametric models of geometry and illumination. IEEE Transactions on Pattern Analysis and Machine Intelligence **20** (1998) 1025–1039