Proc. of the 23st DAGM Symposium für Mustererkennung, Springer Verlag, 2001.

Learning shape models from examples

D. M. Gavrila[1] and J. Giebel[1] and H. Neumann[2]

[1] Image Understanding Systems, DaimlerChrysler Research,
Wilhelm Runge St 11, 89081 Ulm, Germany
dariu.gavrila, jan.giebel@DaimlerChrysler.com, www.gavrila.net
[2] Department of Neuro-informatics, University of Ulm,
89069 Ulm, Germany
hneumann@neuro.informatik.uni-ulm.de

Abstract. This paper addresses the problem of learning shape models from examples. The contributions are twofold. First, a comparative study is performed of various methods for establishing shape correspondence - based on shape decomposition, feature selection and alignment. Various registration methods using polygonal and Fourier features are extended to deal with shapes at multiple scales and the importance of doing so is illustrated. Second, we consider an appearance-based modeling technique which represents a shape distribution in terms of clusters containing similar shapes; each cluster is associated with a separate feature space. This representation is obtained by applying a novel simultaneous shape registration and clustering procedure on a set of training shapes. We illustrate the various techniques on pedestrian and plane shapes.

1 Introduction

For many interesting object detection tasks there are no explicit prior models available to support a matching process. This is typically the case for the detection of complex non-rigid objects under unrestricted viewpoints and/or under changing illumination conditions. In this paper we deal with methods for acquiring (i.e. "learning") shape models from examples. Section 2 reviews existing methods for establishing shape correspondence and modeling shape variation. Shape registration methods bypass the need for tedious manual labeling and establish point correspondences between shapes in a training set automatically. Although a sizeable literature exists in this area (e.g. [1] [2] [3] [5] [9]), there has been little effort done so far in comparing various approaches. We perform a comparative study on various shape registration methods in Section 3 and demonstrate the benefit of describing shapes at multiple scales for this purpose. The best performing registration method is combined with a clustering algorithm to describe arbitrary shape distributions in terms of clusters containing similar shapes; each cluster is associated with a separate feature space, see Section 4. Benefits of such representation are discussed in Section 5, after which we conclude in Section 6.

2 Review

2.1 Shape Registration

A review of previous work on shape registration shows that a typical procedure follows a similar succession of steps: shape decomposition, feature selection, point correspondence and finally, alignment.

The first step, shape decomposition, involves determining control ("landmark") points along a contour and breaking the shape into corresponding segments. One way of doing this is to consider the curvature function along the object's contour and to determine the locations of the minima and maxima. The curvature function is computed by convolving the edge direction function of a contour point with the first derivative of a Gaussian function. The parameter σ in the Gaussian function determines the smoothing of a shape, see for example [11] [12] [13]. A different method for determining control points is described in [14], where points are removed according to a criticality measure based on the area of three successive points.

Once shape segments have been determined, the next step involves selecting features that are transformation invariant (e.g. translation, rotation and scale). These features are usually selected based on an approximation of the shape segments; most common are straight-line (i.e. polygonal) [3] [5] [9] [11] and Fourier approximations [7]. Similarity measures are typically based on length ratios and angles between successive segments for the polygonal case (e.g. [11]) and weighted Euclidean metrics on the low-order Fourier coefficients, for the Fourier approximations (e.g. [7]).

At this point, correspondence between the control points of two shapes can be established by means of either combinatoric approaches or sequential pattern matching techniques. Each correspondence is evaluated using match measures on the local features discussed earlier. Combinatoric approaches [3] [11] select iteratively a (e.g. minimal) set of initial correspondences and use various greedy methods to assign the remaining points. The advantage of this approach is that the overall goodness of a match can be based on all contour points simultaneously. Additional effort is necessary, though, to account for ordering constraints. Sequential pattern matching techniques, on the other hand, inherently enforce ordering constraints. Dynamic programming is widely used [5] [7] since it is an efficient technique to come up with optimal solutions for the case of an evaluation measure which has the Markov property.

Finally, after correspondence between control points has been established (and possibly, between interpolated points), alignment with respect to similarity transformation is achieved by a least-squares fit [2]. The above techniques for shape registration have important applications for partial curve matching in object recognition tasks [5] [7] [11]. Next subsection deals with their use for modeling shape variation.

2.2 Modeling Shape Variation

Registration establishes point correspondence between a pair of shapes. The straightforward way to account for N shapes is to embed all N shapes in a single feature vector space, based on the x and y locations of their corresponding points. This is done either by selecting one shape (typically, the "closest" to the others) and aligning all others to it, or by employing a somewhat more complex hierarchical procedure [9]. The resulting vector space allows the computation of various compact representations for the shape distribution, based on radial (mean-variance) [6] or modal (linear subspace) [1] [2] [9] decompositions. Combinations are also possible [8].

The assumption that point correspondence can be established across all training shapes of a particular object class by means of automatic shape registration is in many cases problematic, however. For example, none of the shape registration methods we analyzed were able to correctly register a pedestrian shape with the two feet apart with one with the feet together, without operator input or prior knowledge. A more general registration approach does not forcibly embed all N shapes into the same feature vector space. Instead, it combines shape clustering and registration, embedding only the (similar) shapes within a cluster into the same vector space. This is the approach pursued in Section 4, adapted from earlier work [3].

Finally, some approaches [4] [5] do not try to embed shapes into a vector space altogether; a hierarchical representation is built solely on the basis of pairwise dissimilarity values. See also Section 5 and [10].

3 Shape Registration

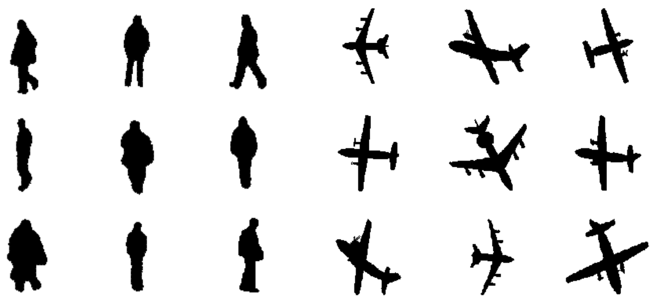
In this study, we jointly compare the performance of control point detection, feature extraction and matching algorithms for the purpose of shape registration. Under consideration are two techniques for control point detection, the first is the Gaussian-based filtering of the curvature function [11] and the second is the critical point detection algorithm described in [14]. We furthermore compare two techniques for feature selection and similarity measurement: one based on polygonal approximations [11] and one based on piecewise Fourier decompositions [7]. We choose as matching algorithm invariably dynamic programming, because of its efficiency and optimality property. This leads to a total of four methods under consideration. The experiments involve computing pairwise alignments between all elements of a particular shape data set and recording the mean Euclidean distance between corresponding contour points after alignment, i.e. the mean alignment error. Dense point correspondences are inferred by interpolating the correspondences between the control points.

Our study furthermore extends previous shape registration work [7] [11] by representing shapes at multiple scales. For the method of [11], this is achieved by using multiple Gaussian σ values (e.g. [12] [13]), whereas in [14] this involves using different area criteria. In the experiments, we compute shape registrations for all scales and consider the one which minimizes the mean alignment error.

Two data sets were used to evaluate the four method combinations, a set of 25 plane contours (height 100 - 200 pixels) and a set of 50 pedestrian contours (height 80 pixels). See Figure 1. Figure 2b illustrates the multi-scale shape representation resulting from the Gaussian filtering of the curvature function. Figure 2c shows the effects of varying the number of Fourier coefficients, at one particular scale.

The importance of maintaining shape representations at multiple scales is demonstrated in Figure 3. The histogram shows the number of alignments (y-axis) at which a particular scale σ-interval (x-axis) results in the best pairwise alignment in terms of minimizing the mean alignment error. This particular data pertains to the registration method that combines control point detection by Gaussian curvature filtering and the use of Fourier features for the pedestrian data set. From the Figure one observes that a wide range of scales are utilized for the "best" alignment; a registration approach that would only use a single scale representation (i.e. choosing a single bar of the histogram) would be non-optimal for most of the shape alignments considered.

Some typical point correspondences and alignments are shown in Figure 4. As can be seen, the registration method is quite successful in pairing up the physically corresponding points (e.g. the tip and wings of the planes, the heads and feet of the pedestrians).



(a) (b)

Fig. 1. Example (a) pedestrian shapes (b) plane shapes.

Figure 5 summarizes the results of the shape registration experiments. It shows the cumulative distribution of the mean alignment error for the plane (Figure 5a) and pedestrian (Figure 5b) data sets, for the four combinations analyzed and based on 600 and 2450 pairwise alignments, respectively. For example, about 80% of the pedestrian alignments resulted in a mean alignment error smaller than 9 pixels (on a pedestrian size of 80 pixels) for method **c-**, Gaussian curvature filtering with polygonal features. More revealing is relative performance of the methods. From the Figure one observes that Gaussian filtering of the curvature function provides a better multi-scale shape representation

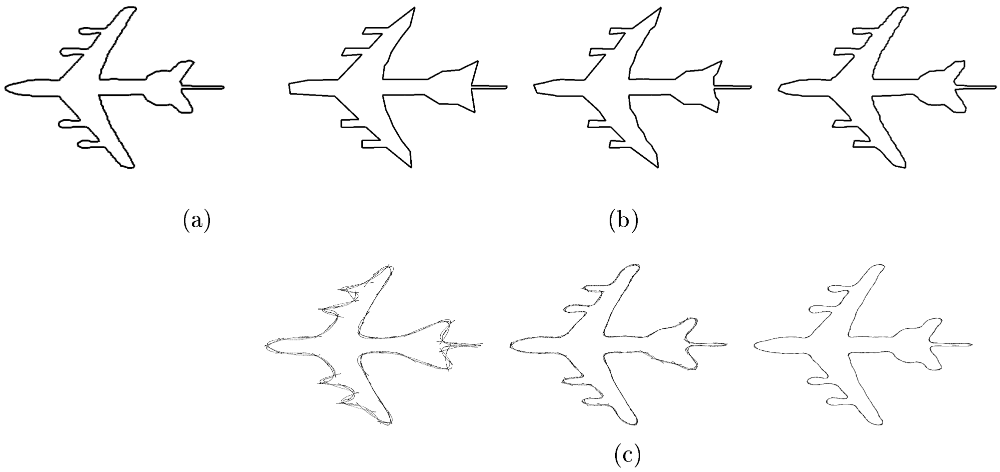


Fig. 2. Multi scale representation (a) original shape (b) polygonal approximations (decreasing σ value from left to right) (c) Fourier approximations (increasing number of coefficients used from left to right)

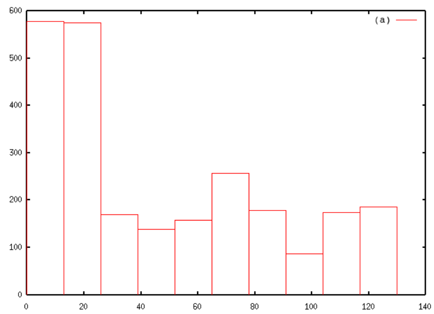
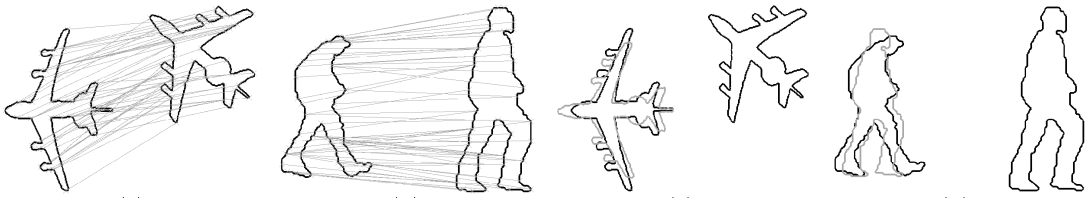


Fig. 3. Histogram showing number of alignments (y-axis) at which a particular scale σ-interval (x-axis) results in the best pairwise alignment, using Gaussian curvature filtering [11]

than that derived from the critical point detection algorithm (compare graphs **a-** and **c-** versus graphs **b-** and **d-**), at least as far as our implementation is concerned. Also, the Fourier features proved to be more suitable in dealing with the curved shapes of our data sets (compare graphs **a-** and **b-** versus graphs **c-** and **d-**).

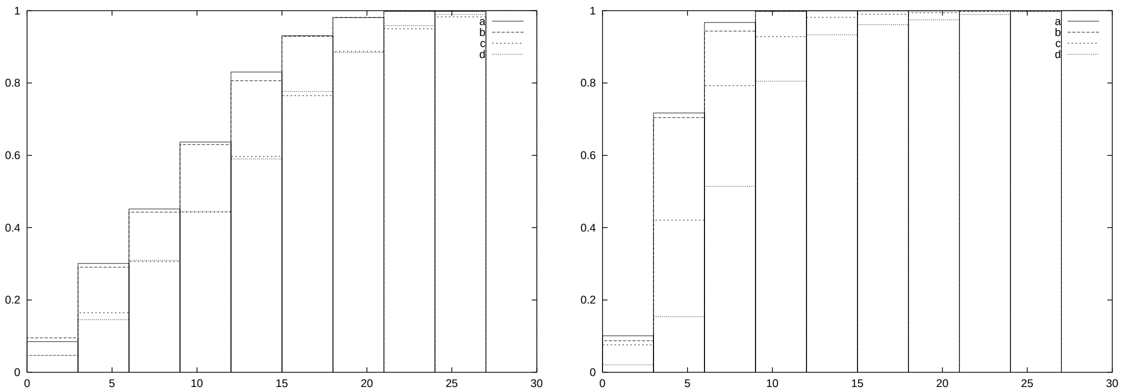
4 Shape Clustering

After the comparative study on shape registration, we used the best performing method (i.e. Gaussian curvature filtering for control point detection and Fourier coefficients as features) to embed the N shapes of the training set into feature spaces. As mentioned in Subsection 2.2, for many datasets it is not feasible to



(c) (d) (c) (d)

Fig. 4. Established point correspondences between two (a) planes and (b) pedestrians and their pairwise alignment, (c) and (d). Aligned shapes are superimposed in grey.



(a) (b)

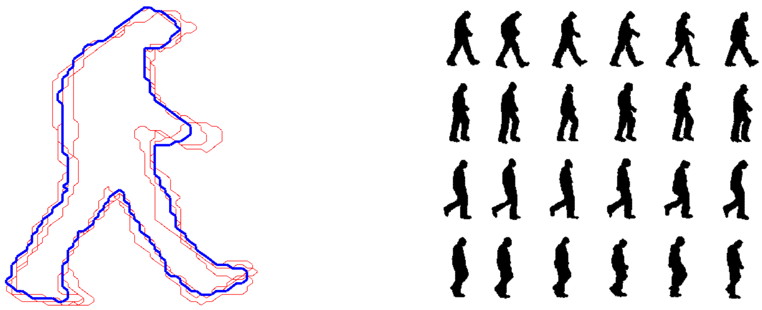
Fig. 5. Cumulative distribution of the mean alignment error for the (a) plane and (b) pedestrian data of the four methods analyzed (**a-** Gaussian curvature filtering [11] multi-scale with Fourier features [7], **b-** "critical point" [14] multi-scale with Fourier features [7], **c-** Gaussian curvature filtering [11] multi-scale with polygonal features [11], and **d-** "critical point" [14] multi-scale with polygonal features [11]). The horizontal axis is in units of pixels. Object size 100 - 200 pixels for (a) and 80 pixels for (b)

map all N shapes onto a single feature space because of considerable shape differences. Therefore, we follow a combined clustering and registration approach which establishes shape correspondence only between (similar) shapes within a cluster. The clustering algorithm is similar to the classical K-means approach:

0. pick an initial shape S_1 and add it to cluster C_1
  as prototype: $C_1 = \{S_1\}$, $P_1 = S_1$
while there are shapes left do
  1. select one of remaining shapes: S_k
  2. compute mean alignment error $d(S_k, P_i)$ from
    element S_k to the existing prototypes P_i,
    where i ranges over the number of clusters
    created so far
  3. Compute $d_{min} = d(S_k, P_j) = min_i d(S_k, P_i)$.
    if $d(S_k, P_j) > \theta$

then assign S_k to a new cluster C_{n+1}:
$C_{n+1} = \{S_k\}, P_{n+1} = S_k$
else assign S_k to existing cluster
$C_j = \{S_{j1}, ...S_{jn}\}$ and update its prototype:
$C_j = C_j \cup \{S_k\}$
$P_j = \text{Mean}(S_{j1}, ..., S_{jn}, S_k)$
end

In the above, Step 2 consists of applying our best performing shape registration method (see previous Section) to establish point correspondences and compute alignment errors. The resulting point correspondences are used for the prototype computation in Step 3. See Figure 6a. Parameter θ is a user-defined dissimilarity threshold that controls the number of clusters to be created. Compared to [3], the above formulation has the advantage that it does not require the computation of the full dissimilarity matrix $d(S_i, S_j)$. It furthermore adjusts the prototype whenever new elements are assigned to a group. Figure 6b illustrates some typical clustering results.



(a) (b)

Fig. 6. (a) Computing a "mean" or prototype shape (in black) from aligned shape samples (in grey) (b) Shape clustering: each row contains the elements of a different cluster

5 Outlook

Learned shape models can be used for matching and tracking purposes. We are currently utilizing the approach described in previous Section for learning pedestrian shapes and improving the performance of the Chamfer System [4], a system for shape-based object detection. At its core, the system performs template matching using distance transforms. It has an off-line phase where a template hierarchy is built by clustering shapes recursively; clustering is solely based on dissimilarity values [4]. Here, the followed shape registration and modeling approach establishes feature spaces which facilitate the computation of improved shape prototypes, i.e. the arithmetic mean of the shape cluster elements. Furthermore, the presence of a feature space enables standard data dimensionality reduction techniques which can used for the generation of "virtual" shapes, enriching the training set of the Chamfer System.

6 Conclusions

This paper performed a comparative study of various methods for shape registration and identified a multi-scale method based on Gaussian curvature filtering and Fourier features as best performing one. This registration step was incorporated in a shape clustering algorithm which allows the representation of arbitrary shape distributions in terms of clusters of similar shapes, embedded in separate feature spaces.

References

1. A. Baumberg and D. Hogg. Learning flexible models from image sequences. In *Proc. of the ECCV*, pages 299–308, 1994.
2. T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models - their training and applications. *CVIU*, 61(1):38–59, 1995.
3. N. Duta, A. Jain, and M.-P. Dubuisson-Jolly. Learning 2d shape models. In *Proc. of the ICCV*, pages 8–14, Kerkyra, Greece, 1999.
4. D. M. Gavrila and V. Philomin. Real-time object detection for "smart" vehicles. In *Proc. of the ICCV*, pages 87–93, Kerkyra, Greece, 1999.
5. Y. Gdalyahu and D. Weinshall. Flexible syntatic matching of curves and its application to automatic hierarchical classification of silhouettes. *IEEE Trans. on PAMI*, 21(12):1312–1328, December 1999.
6. C. Goodall. Procrustes methods in the statistical analysis of shape. *J. Royal Statistical Soc. B*, 53(2):285–339, 1991.
7. J. Gorman, R. Mitchell, and F. Kuhl. Partial shape recognition using dynamic programming. *IEEE Trans. on PAMI*, 10(2):257–266, 1988.
8. T. Heap and D. Hogg. Improving the specificity in PDMs using a hierarchical approach. In *Proc. of the BMVC*, Colchester, UK, 1997. BMVA Press.
9. A. Hill, C. Taylor, and A. Brett. A framework for automatic landmark identification using a new method of nonrigid correspondence. *IEEE Trans. on PAMI*, 22(3):241–251, March 2000.
10. D. Jacobs, D. Weinshall, and Y. Gdalyahu. Condensing image databases when retrieval is based on non-metric distances. In *Proc. of the IEEE CVPR Conf.*, pages 596–601, Ft. Collins, U.S.A., 1999.
11. H.-C. Liu and M. Srinath. Partial shape classification using contour matching in distance transformation. *IEEE Trans. on PAMI*, 12(11):1072–1079, November 1990.
12. F. Mokhtarian and A. K. Mackworth. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Trans. on PAMI*, 14(8):789–805, 1992.
13. A. Rattarangsi and R. T. Chin. Scale-based detection of corners of planar curves. *IEEE Trans. on PAMI*, 14(4):430–449, 1992.
14. P. Zhu and P. Chirlian. On critical point detection of digital shape. *IEEE Trans. on PAMI*, 17(8):737–748, 1995.