Convex Layers: A New Tool for Recognition of Projectively Deformed Point Sets

Tomáš Suk and Jan Flusser *

Institute of Information Theory and Automation
Academy of Sciences of the Czech Republic
Pod vodárenskou věží 4, 182 08 Praha 8, Czech Republic
suk@utia.cas.cz, flusser@utia.cas.cz

Abstract. A new more efficient algorithm for recognition of projectively deformed point sets is presented. It supposes the type of projective transformation, where the orientation of a triangle is preserved.
First, convex layers of two compared sets are computed and left-tangents are found. If their structures are different, then an advanced algorithm is used. It creates the new structure of one set according to the other set. If the numbers of the points in the convex layers differ from each other, then the suitable points are moved to other layers, and if the left-tangents do not link the corresponding points, then their structure is corrected. The suitable combination of optimal structure correction criteria was investigated by a numerical experiment, the angle criteria were found as the best.
Finally, five-point cross ratios of sequences of the points from both sets are computed and compared.

1 Introduction

One of the important tasks in image processing and computer vision is recognition of objects on images captured under different viewing angles. If the scene is planar, then the distortion between two frames can be described by *projective transform*

$$\begin{aligned} x' &= (a_0 + a_1 x + a_2 y)/(1 + c_1 x + c_2 y) \\ y' &= (b_0 + b_1 x + b_2 y)/(1 + c_1 x + c_2 y)\ , \end{aligned} \tag{1}$$

where x, y are the coordinates in the first frame and x', y' are the coordinates in the second one.

The scope of this contribution is limited to the cases, where objects on images can be described by finite sets of points and its projective transforms have th same number of points, e.g. vertices of polygons. The algorithm does not solve cases, when some points are occluded on some images.

Under (1) the straight line

$$1 + c_1 x + c_2 y = 0 \tag{2}$$

* This work has been supported by the grant No. 102/98/P069 of the Grant Agency of the Czech Republic.

is not mapped into the second plane (or it is mapped into the infinity). Each triangle not intersected by the straight line (2) is transformed by (1) with the same orientation of the vertices. It implies some additional theorems dealing with topology of the set, e.g. the convex hull is preserved during the projective transform in that case. This property can be used for construction of very efficient algorithms for recognition of point sets. While the computing complexity of the invariants [2] in the general case of the projective transform when the points can lay in both parts of the plane is $O(n^5)$, there was published an algorithm for recognition of projectively deformed point sets in [1], which utilizes preserving triangle orientation, with computing complexity $O(n \log n)$.

Concisely, it consists of a few steps:

1. Compute convex layers of both sets.
2. Compute left-tangents.
3. Acquire left-list of points.
4. If the structures of the sets are the same, compute five-point cross ratios and declare the sets as equivalent if and only if the cross ratios are equivalent.

The convex layers of a set are obtained by computing convex hull of the set, removing the points of the hull and computing convex hull of the rest of the set. It is repeated while some points remain in the set.

From a point of the outer layer we can draw the two tangents to the adjacent inner convex layer. While looking at the inner hull from the point, we can classify these tangents as left-tangents and right-tangents (Fig. 1 a). We must choose one of them, but the other can be used as well. The left-tangents were chosen here.

The system of the left-tangents with fictitious point in the center of the innermost convex layer forms a tree. The tree can be transcribed into a sequence of the points by depth-first search of the tree. The sequence is called left-list. The transcription is ambiguous, we can choose any point in the innermost hull as the beginning. If the innermost hull contains one point only, the same situation occurs in the next convex layer. We must consider each of these sequences and find the pair of sequences, one from the first set and the other from the other set, with the least deviation of the cross ratios. If the deviation is zero, then the sets are precisely equivalent, else the deviation can be considered as the measure of the degree of difference between the sets.

The main disadvantage of this algorithm is its high sensitivity to noise. If some three points are almost collinear, then small perturbations in their coordinates can affect the structures of the convex hulls and left-tangents and the sets are declared as completely different. This paper describes a new algorithm which is more robust to noise in point coordinates.

The simplest projective invariant is a five-point cross ratio [3, 4, 5 and 6]:

$$\varrho(A,B,C,D,E) = \frac{P(A,B,C)P(A,D,E)}{P(A,D,C)P(A,B,E)} \tag{3}$$

where $P(A,B,C)$ is the area of the triangle with vertices A, B and C. The point A is called the common point of the cross ratio.

Since the set of n points has $2n$ degrees of freedom and the projective transform has 8 parameters, therefore the set can have only

$$m = 2n - 8 \tag{4}$$

independent invariants to projective transform. In [1] there are only n invariants computed for n points, we propose to take m invariants, which describe the set completely. If we have a sequence of n points, we can compute the cross ratio of the first point as the common one and following 4 points, then the second point as the common one and following 4 points and so on until we take the fifth point from the end as the common one and last 4 points. We obtain $m/2$ invariants. Now we take the sequence in the reverse order, it means the last point as the common one and 4 preceding points, then last but one as common and preceding 4 points and so on until the fifth point as common one and first 4 points. So each point is used at least once as the common one in m cross ratios.

Since the cross ratio can be infinity or near infinity and it can lead to instability, we use the arc tangent of the cross ratio instead of the mere cross ratio.

2 Basic algorithm

Our algorithm determines, if two point sets are equivalent up to projective transform. It can be outlined as follows

1. Compute convex layers of both sets.
2. Compute left-tangents.
3. Obtain left-list of points.
4. Find the degree of symmetry of the left-list.
5. If the structures of the sets differ from each other, use advanced algorithm from the next section.
6. For each degree of symmetry compute invariants of the corresponding sequence.
7. The minimum distance of two sets of the invariants in Euclidean feature space expresses the measure of equivalence of the point sets.

The convex layers can be computed either the Chazelle's fast algorithm [7] with the computing complexity $O(n \log n)$ or much simpler Jarvis' march [8] with running time $O(n^2)$. If we know the convex layers, then the left-tangents can be found with the computing complexity $O(n)$.

The tree is created from the left-tangents and transcribed into the left-list by the depth-first search of the tree (Fig. 1 c). There are the convex layers outlined by the thick line and the left-tangents by the thin line in the Fig. 1 b.

The left-list of this set can be

$$C_1, B_1, A_1, A_2, C_2, B_2, A_3, B_3, A_4, C_3, B_4, A_5.$$

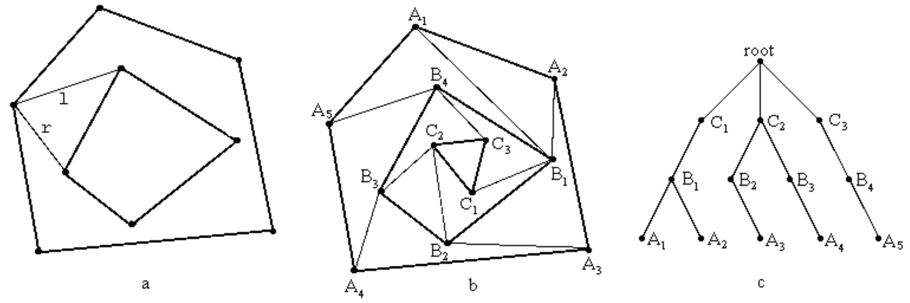

Fig. 1. a: Left-tangent (l) and right-tangent (r). b: The convex layers (thick line) and the left-tangents (thin line). c: The tree from the left-tangents

From all possible sequences is used that one with the greatest lexicographic order. If we want to find which of two sequences has greater lexicographic order, we find the first points from the beginnings, whose convex depths differ from each other, and the sequence with the greater convex depth of the point is lexicographically greater. E.g. the sequence of the convex depths of the left-list from the Figs. 1 b,c is

$$3, 2, 1, 1, 3, 2, 1, 2, 1, 3, 2, 1.$$

We can use all cyclic permutations beginning by 3, but we use the sequence

$$3, 2, 1, 3, 2, 1, 1, 3, 2, 1, 2, 1$$

because it is lexicographically greatest – the fourth member of this permutation is the greatest.

In case of only one point in the innermost convex layer, this point is ever first in the left-list and it is omitted from the cyclic permutations.

The degree of symmetry is the number expressing how many cyclic permutations of a sequence is the same as the original sequence. For instance, the degree of symmetry of the sequence

$$3, 2, 1, 1, 3, 2, 1, 1, 3, 2, 1, 1$$

equals 3. The degree of symmetry of a sequence without any symmetry equals 1. If the degree of symmetry of the left-list depths is greater than 1, then we must consider the corresponding number of the cyclic permutations.

The structures of both sets are identical, if the sequence of the convex depths of the points in the left-lists are the same, otherwise the advanced algorithm must be used. It is described in the next section.

The minimum distance of two sets of the invariants in m-dimensional Euclidean feature space expresses the measure of equivalence of the point sets. The sets are exactly equivalent, it means they can be transformed by projective transform one into another, if and only if their distance equals zero. The number of structural changes can serve as the auxiliary criterion of similarity.

3 Advanced Algorithm

The aim of this algorithm is to arrange the structure of one point set according to the structure of the other set, what is refining with respect to the basic algorithm, which cannot change the structure.

The advanced algorithm consists of two steps, translations of points into different convex layers (1,2) and arrangement of left-tangents (3,4). First, one set is labeled as reference and the other one as input.

Data from the basic algorithm:
n = number of points in one set
$cd_1[i]$ = convex depth of the i-th point of the input set
$cd_2[i]$ = convex depth of the i-th point of the reference set
nd = convex depth of of the reference set
$np[i]$ = number of points with convex depth = i in the reference set
$lt_1[i,k]$ = left-tangent of the k-th point in the i-th layer (input set)
$lt_2[i,k]$ = left-tangent of the k-th point in the i-th layer (reference set)
Left-tangent means its order in its convex layer.

Advanced algorithm:

1. for i:=1 to n
   if($cd_1[i] < cd_2[i]$)
   $point\_down(cd_1[i], cd_2[i])$
   if($cd_1[i] > cd_2[i]$)
   $point\_up(cd_1[i], cd_2[i])$
   endfor(i)

2. for i:=1 to n
   if the i-th point was moved in Step 1
   $inner\_tangent(lt_1, i)$
   endfor(i)

3. for j:=1 to $np[1]$ //loop via points of the first layer
   for i:=1 to $nd - 1$ //loop via other layers
   for k:=1 to $np[i]$ //loop via points of the layer
   $no := (lt_1[i,k] - lt_2[i,k]) \bmod np[i]$
   $sh[no] := sh[no] + 1$ //compute histogram of the layer rotations
   endfor(k)
   Find maximum $maxr$ of the histogram sh.
   If there is more identical maxima, compute sum of left-tangent criteria for each of them.
   $noerl := noerl + np[i] - maxr$ //number of changes of the left-tangents
   endfor(i)
   $minnoerl$:=minimum number of changes $noerl$
   If there is more identical minima, we consider that with less sum of the left-tangent criteria.
   $nchm[i]$:=position of the minimum

endfor(j)

4. for i:=1 to nd
   for j:=1 to $np[i]$
   $lt_1[i,j] := lt_2[i,(j+nchm[i]) \bmod np[i]]$
   endfor(j)
   endfor(i)

The procedures $point\_down(i,j)$ and $point\_up(i,j)$ search, what point from the i-th layer is the best candidate for translation to the j-th layer. The procedure $point\_up(i,j)$ examines triangles from three adjacent points in the source layer, the procedure $point\_down(i,j)$ examines triangles from a point in the source layer and two corresponding points in the destination layer [1]. A number of both the layer criteria and the tangent criteria was tested, see the section Numerical Experiments.

The procedure $inner\_tangent(lt,i)$ compute a new left-tangent of the translated i-th point.

The aim of the next part of the algorithm is to find the optimal way, how to copy left-tanget numbers from the reference set to the input one. The criterion is the whole number of changes and if there is more than one identical minima, we use another auxiliary left-tangent criterion.

4 Numerical Experiments

The following two numerical experiments were carried out. The goal of the first experiment was to determine, what layer and left-tangent criteria are the best.

A set of 16 points was created by a pseudo-random generator. They are uniformly distributed from 0 to 511. Independent zero-mean Gaussian noise with gradually increasing standard deviation up to 30 was added to the coordinates of all points and the original set was compared with the noisy set by our algorithm. The experiment was repeated 100 times for each standard deviation and for each of 7 combinations of the criteria. The following combinations of the criteria and the whole number of correct results can be seen in Tab. 1. The criteria are illustrated in Fig. 2.

We can see the success rate of the 7th combination is the best one. The success rate of the 6th combination is only slightly worse.

The goal of the second experiment was the comparison of the advanced algorithm with mere base algorithm.

Two basic sets of 16 uniformly distributed points were created again. One of them was corrupted by noise like in the previous case. The noisy set was compared both with the original set and with the other set by our algorithm.

[1] The points are corresponding, when the point in the source layer lies in the triangle from two adjacent points in the destination layer and the centroid of the innermost layer, e.g. the pair of the points A_3 and A_4 is corresponding to the point B_2 in the Fig. 1 b.

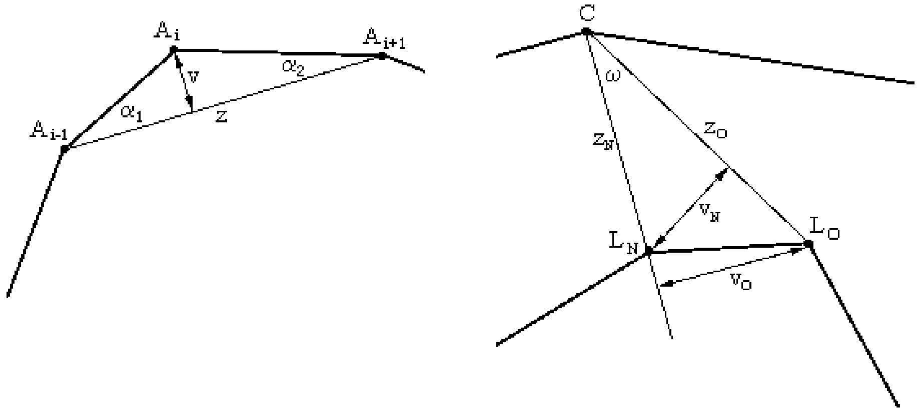

Fig. 2. The criteria of the point translations to another layer (left) and those of the new left-tangent (right)

10 basic sets were created for each standard deviation and the experiment was repeated 10 times for each pair of the basic sets and for each standard deviation. The average deviations in the feature space can be seen in Fig. 3 left. The solid line expresses comparison with the original set and the dashed line with the other set. The number of the correct results can be seen in Fig. 3 right. The solid line expresses the number of correct results of the advanced algorithm and the dashed line that of the basic algorithm.

Table 1: The combinations of the criteria

No.	layer criterion	left-tangent criterion	correct
1	distance v	old distance v_O	739
2	relative distance v/z	relative old distance v_O/z_N	724
3	distance v	new distance v_N	740
4	relative distance v/z	relative new distance v_N/z_O	723
5	triangle area $P = vz/2$	triangle area $P_l = v_O z_N/2 = v_N z_O/2$	746
6	angle maximum of α_1 and α_2	angle ω	854
7	angle sum $\alpha_1 + \alpha_2$	angle ω	860

We can see that the advanced algorithm yields good results in case of significantly greater noise than the mere basic algorithm without possibility of structure corrections.

5 Conclusion

In this contribution, we improved the original method for recognition of projectively deformed point sets [1]. The simpler evaluation of symmetric sets was

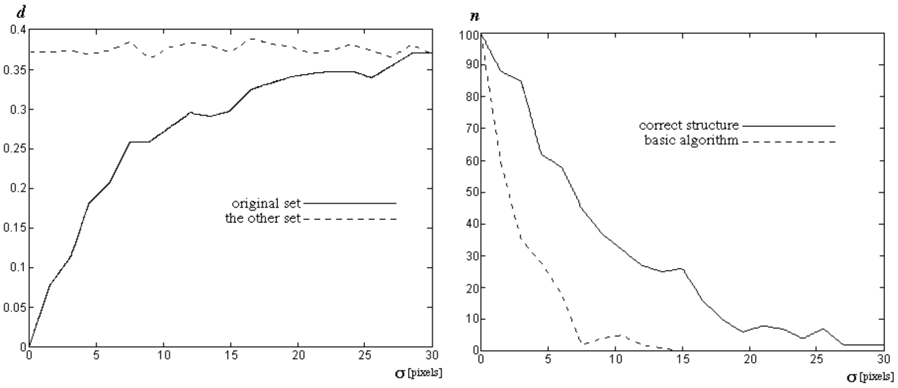

Fig. 3. The average deviations d in the feature space (left) and the number n of the correct results (right) with respect to the noise standard deviation σ

used instead of relatively complicated searching substrings in the string and more suitable number of invariants was computed. The new method of structure correction was established. The correspondence between the points of both sets can be found in case of need. The behavior of the method was demonstrated by numerical experiments, one of them showed the angle criteria of the moving point are the most suitable.

References

1. Rao N. S. V., Wu W., Glover C. W.: Algorithms for Recognizing Planar Polygonal Configurations Using Perspective Images, In IEEE Trans. on Robotics and Automation, Vol. 8 (1992), No. 4, pp. 480-486
2. Suk T., Flusser J.: The Features for Recognition of Projectively Deformed Images, In Proc. ICIP'95, Washington (1995), Vol. III, pp. 348-351
3. Reiss T. H.: Recognition Planar Objects Using Invariant Image Features, Lecture Notes in Computer Science, Vol. 676, Springer (1993)
4. Lenz R., Meer P.: Point configuration invariants under simultaneous projective and permutation transformations, In Pattern Recognition, Vol. 27 (1994), No. 11, pp. 1523-1532
5. Maybank S. J.: Classification Based on the Cross Ratio, In proc. of The Darpa-Esprit Workshop on Applications of Invariants, Azores, Portugal (1993), pp. 113-132
6. Sinclair D., Boufama B., Mohr R.: Independent Motion Segmentation and Collision Prediction for Road Vehicles, In COPR94 (1994), pp. 958-961
7. Chazelle B.: On the Convex Layers of a Planar Set, In IEEE Trans. on Information Theory, Vol. IT-31 (1985), No. 4, pp. 509-517
8. Preparata F. P., Shamos M. I.: Computational Geometry, Springer (1985)