

LWDOS: Language for Writing Descriptors of Outline Shapes

Slimane Larabi¹, Saliha Bouagar¹, Felix Miguel Trespaderne², and
Eusebio de la Fuente Lopez²

¹ Computer Science Institute of U.S.T.H.B. University
BP 32, El-Alia, Algiers, Algeria
larabi@wissal.dz

² Departamento Ingeniera de Sistemas y Automtica ETS Ingenieros Industriales
P. del Cauce s/n 47011 Valladolid , Spain
{trespa,eusfue}@eis.uva.es

Abstract. A new method to obtain a rough description of a 3-D object from its outline shape is presented in this paper. Firstly the outline shape is split up into parts that are related in a connectivity graph. The boundary features of every part (lines, curves) and the information about its junction with other parts are extracted in order to provide a semantic content to the outline shape graph. A specific language to describe this graph has been developed. Finally, some results obtained applying our approach over real images are presented and discussed

1 Introduction

The development of reliable object recognition systems is specially important in a wide variety of problems, such as image classification, content-based image retrieval, tracking and surveillance. Although much research has been done in this direction, the problem remains unsolved. In order to recognize objects, a model database containing all reference images of the objects is stored. The database contains a specific description of objects and it is indexed to reduce the time of comparison process. Several representations has been presented in the literature to describe the image of a free form 3D object. These representations are based on:

- Appearance that consist to code the image of the object directly without a previous segmentation process. Principal component analysis on a set of training image data, noted eigenspace [5, 6, 22, 26], is used.
- Parts (volumic primitives, or 2D parts) where a part contains a rich information extracted from images such as small image patches [1, 4, 14, 20]
- Boundary curves of silhouette by calculating their representation using the centroidal profile, curvature-tuned smoothing (CTS) or curvature scale space (CSS) [3, 10, 15, 21, 28]
- Features or invariant features extracted from boundaries of the silhouette of the object [7, 13, 16, 17, 19, 23]

- 3D boundaries reconstructed by segment-based stereo vision [27]
- A set of perceptual dimensions (naturalness, openness, roughness, expansion and ruggedness) representing the dominant spatial structure of a scene [24]

The outline shape information, defined simply as the boundary contour of object from a particular viewpoint, has been very used in recognition tasks. Its sufficiency for models in object recognition has been investigated by W. G. Hayward [12] concluding that silhouettes provide useful information for recognition processes. In this paper the outline shape is exploited to calculate a rough description of the correspondent 3-D object. Firstly the outline shape is split up into parts that are associated in a connectivity graph. A semantic is added to this graph extracting for each part, features of its boundaries (lines, curves) and information about its junction with other parts. A specific language has been developed to describe this graph. The main contributions of our method are:

- The description of the outline shape takes into account all morphological details: curvature, relative length, area and disposition
- The proposition of a specific language LWDOS to write linear descriptors, invariant to the scale change and rotation

In following sections, the modeling of the outline shape is first presented and afterwards the developed language LWDOS. Finally the experiments conducted over real images and their results are discussed.

2 Modeling of 3-D object from outline shape

2.1 Notion of part

In order to recognize three-dimensional objects using images it is necessary to find a description of 3D objects from images that must be invariant to scale change and rotation. In this study, we suppose that outline shapes of objects can be located on the image using features extractions techniques.

Let (Oxy) be the referential attached to the minimum rectangle MR that encloses the outline shape OS , chosen such as the origin O is the left top edge of MR and the Ox (resp. Oy) axis corresponds to the width (resp. length) of OS (see Figure 1.a). We define a x-part (resp. y-part) as a set of successive x-rows (resp. y-rows) of silhouette pixels where a x-row (resp. y-row) corresponds to the set of contiguous pixels having the Ox (resp. Oy) direction (see Figure 1.a).

Any outline shape can be split into parts following the Ox or Oy direction. For example, seven x-parts compose the outline shape of the figure 1. Three different x-parts are attached to a fourth x-part, which is attached to three other x-parts.

Extracting features of parts Every part is composed by two boundaries (left and right)(see figure 2). The pixel (or the set of pixels) at the top of the part is

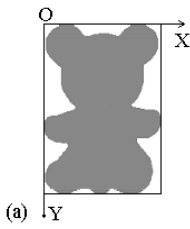


Fig. 1. The split process

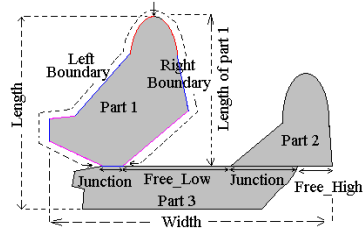


Fig. 2. Geometry of parts

the beginning point (or the first line segment) of the two boundaries. The contour of every boundary will be split up into features (segments, curves) following the technique proposed by G. Dudek and J. K. Tsotos [10]. For example, the left boundary of part 1 of the figure 2 is split up into one convex curve and four line segments. The right boundary is split up into one convex curve and two line segments

2.2 Representation of outline shape from parts

Preliminary representation The outline shape can be described with a graph [8] where the nodes correspond to the parts and the arcs represent the attachment relation between them. This graph needs some semantic information in order to eliminate the ambiguity from the graphs associated to outline shapes. Otherwise the same description can be matched with different outline shapes. Thus, it is necessary to insert for every arc and node a set of characteristics that allow to eliminate the ambiguity.

Geometry of boundary features To describe shapes we will use qualitative terms rather numerical values [8, 11, 25]. To each boundary part it will be associated the description and relative size of its features. Any feature may be a line segment, convex curve, or concave curve. The attributes: weakly convex (*Wcv*), convex (*Cv*), strongly convex (*Scv*), weakly concave (*Wcc*), concave (*Cc*), strongly concave (*Scc*) and line (*Ln*) will be used to characterize the feature form. Another attribute is added to this description to characterize the direction of feature inclination (to left or to right). Analogously it will be associated the attributes horizontal (*Hor*), vertical (*Vrt*), oriented to left (*Ol*), oriented to right (*Or*), weakly inclined to left (*Wil*), inclined to left (*Il*), strongly inclined to left (*Sil*), weakly inclined to right (*Wir*), inclined to right (*Ir*) and strongly inclined to right (*Sir*) to characterize its orientation. The relative size of every feature is characterized indicating its relative length in relation to the outline shape length. Instead to use the numerical values of calculated ratios, the attributes (*VeryShort*), (*Short*), (*LessShort*), (*Middle*), (*EnoughLong*), (*Long*), (*VeryLong*), (*Entire*) are used for the quantification of lengths. Finally, we will

use the notation:

$(FeatureForm)$ $(FeatureOrientation)$ $(RelativeSize)$ to write the geometric description of any feature. For example $(WcvIlShort)$ designs a short curve weakly convex and inclined to left.

Geometry of the junction, disjunction line Every transition between parts must be described indicating how the parts are joined. Any junction or disjunction line is characterized by a set of segments. Every segment must incorporate information about whether it corresponds to a *Junction* (in case where the segment is common for two parts), or it is free (in case where the segment belongs only to the High (*FreeHigh*) or Low part (*FreeLow*)). We note for this: $JunctLine = \{(state, ReLength)^+\}$, where *state* takes one of *Junction*, *FreeHigh*, *FreeLow* values, *ReLength* denotes the relative length of the junction segment.

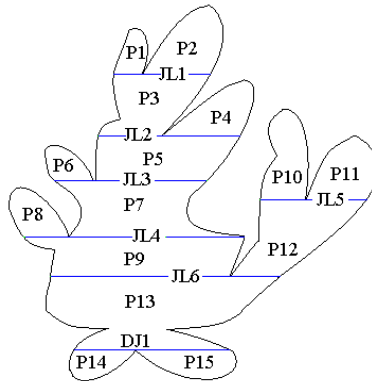


Fig. 3. Composed parts

3 Language to writing descriptors

3.1 Few notions

A specific language noted LWDOS (Language for Writing Descriptors of Outline Shapes) have been developed in order to translate into descriptors the graph associated to any outline shape. In the literature, the languages proposed are only valid for sketches [2, 9, 18, 25]. We will use the following notations to define the LWDOS Language:

- *JuncLine* (resp; *DisjLine*) refers to the junction (resp. disjunction) line
- *UnitedPart* refers to the set of parts that are joined with a part through the junction line

- *ComposedPart* refers to the set of *UnitedPart*, *JunctLine* or *DisjLine* and the parts joined
- *LeftBound* (resp. *RightBound*) refers to the left (resp. right) boundary
- *ContDescriptor* refers to the descriptor of contour
- *ContGeometry* refers to the geometry of contour

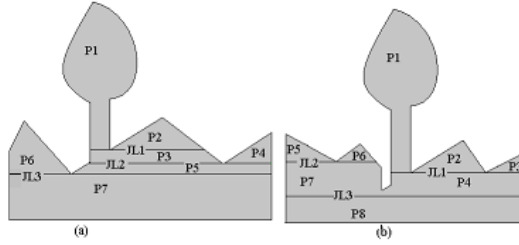


Fig. 4. Outline shapes

3.2 Definition of the LWDOS language

The LWDOS grammar is given by the 4-tuple $LWDOS = (V_N, V_T, P, S_0)$ where: V_T, V_N are respectively the finite set of terminal vocabulary and the finite set of non-terminal vocabulary, $S_0 \in V_N$ is the starting symbol, and P is a finite set of production rules of the type $\alpha \rightarrow \beta$, where $\alpha \in V_N$ and $\beta \in (V_N \cup V_T)^*$ of all string. The non-terminal vocabulary of LWDOS language is written as:

$V_N = \{S_0, ComposedPart, UnitedPart, Part, LeftBound, RightBound, ContDescriptor, ContGeometry, JunctLine, DisjLine, State, ReLength\}$

The terminal vocabulary of LWDOS language is written as:

$V_T = \{ \cup, \oplus, \uparrow, \downarrow, [,], \{, \}, Junction, FreeHigh, FreeLow, VeryShort, Short, LessShort, Middle, EnoughLong, Long, VeryLong, Entire \} \cup Set_attr_feat$

Where $Set_Attr_Feat = \{LnVrt, WcvVrt, CvVrt, ScvVrt, WccVrt, CcVrt, ScvVrt, LnHorOl, WcvHorOl, \dots ScvHorOl, \dots\}$

$\{LnSir, WcvSir, CvSir, ScvSir, WccSir, CcSir, ScvSir\}$

The LWDOS production rules P are written as follows:

$S_0 \rightarrow [Part]$

$Part \rightarrow \{LeftBound \oplus RightBound\} / [ComposedPart] / \varepsilon$

$ComposedPart \rightarrow UnitedPart \oplus \uparrow JunctLine \oplus Part / Part \oplus \downarrow DisjLine \oplus UnitedPart$

$UnitedPart \rightarrow Part \cup UnitedPart / Part / \varepsilon$

$LeftBound \rightarrow ContDescriptor LeftBound / ContDescriptor$

$RightBound \rightarrow ContDescriptor RightBound / ContDescriptor$

$ContDescriptor \rightarrow ContGeometry ReLength$

$ContGeometry \rightarrow LnVrt / WcvVrt / \dots / WccSir / CcSir / ScvSir$

JuncLine \longrightarrow *State ReLength/ State ReLength JuncLine*

DisjLine \longrightarrow *State ReLength/ State ReLength DisjLine*

State \longrightarrow *Junction / FreeHigh / FreeLow*

ReLength \longrightarrow *VeryShort/Short/.../VeryLong/Entire*

For example, the LWDOS descriptor of the outline shape of the figure 3 is:

$$[[P_1 \cup P_2 \oplus \uparrow JL_1 \oplus P_3 \cup P_4 \oplus \uparrow JL_2 \oplus P_5 \cup P_6 \oplus \uparrow JL_3 \oplus P_7 \cup P_8 \oplus \uparrow JL_4 \oplus P_9] \cup [P_{10} \cup P_{11} \oplus \uparrow JL_5 \oplus P_{12}] \oplus \uparrow JL_6 \oplus P_{13}] \oplus \downarrow DJL_1 \oplus P_{14} \cup P_{15}]$$

3.3 Utility of the delimiter symbols

A set of delimiter symbols { } [] have been used in the terminal vocabulary in order to facilitate the indexation and comparison processus. For example, it is simple to distinguish between descriptors of the outline shapes (a) and (b) of the figure 4 checking the position of delimiter symbols.

(a) \longrightarrow $[[P_1 \cup P_2 \oplus \uparrow JL_1 \oplus P_3 \cup P_4 \oplus \uparrow JL_2 \oplus P_5 \cup P_6 \oplus \uparrow JL_3 \oplus P_7]$

(b) \longrightarrow $[[P_1 \cup P_2 \cup P_3 \oplus \uparrow JL_1 \oplus P_4] \cup [P_5 \cup P_6 \oplus \uparrow JL_2 \oplus P_7] \oplus \uparrow JL_3 \oplus P_8]$

3.4 Properties of LWDOS descriptors

The properties of LWDOS descriptors can be summarized as follow:

- There is a relation between the two calculated descriptors following the directions bottom-top and top-bottom. One descriptor can be obtained from the other by copying the first descriptor beginning from its end towards the first delimiter and permuting junction line by disjunction line
- Using the relative lengths, the computed descriptor is invariant to any scale change
- Invariance of LWDOS descriptors to rotation is achieved by computing using the referential defined by the minimum rectangle that encloses the outline shape

4 Experiments

4.1 Descriptor's computation

The method has been tested on a set of real images of free form 3D objects acquired in several positions. The object shapes were selected to have significant number of features (see figures 5). The different outline shapes illustrated by figure 6 have been served to test the implanted procedure. We have obtained theses outline shapes applying a binarization technique.

4.2 Descriptor's coding

The computed descriptors of outline shapes has been stored using one byte to code the vocabulary of the LWDOS Language. From conducted experiments, the code of every computed descriptor of outline shape does not exceed 100 bytes. This allows to use one Megabyte to store 10000 different descriptors.

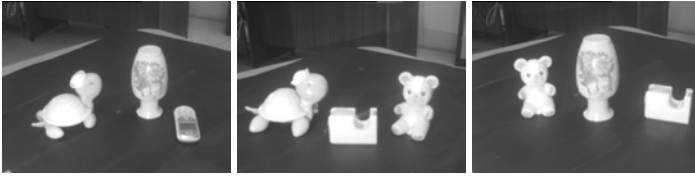


Fig. 5. Set of images test



Fig. 6. The corresponding outline shapes

5 Conclusion and future work

In this paper a geometric modeling of outline shape of free form 3D objects has been presented. We have also proposed a LWDOS language to write the corresponding descriptors that are easy to manipulate and store in a small amount of memory. Our method have been implanted and applied to real images. The results obtained shows the feasibility to store an important number of outline shapes using only few megabytes. However, some problems can be encountered in practice such as outline shape corresponding to several objects, objects having a same regular shape and the noise generating additional parts, junction line or disjunction line. These problems constitute a part of our actual work, as well as the identification process, occlusion and shadow problems. In order to eliminate ambiguity problem, we are working also to incorporate the different regions of the object image in the description of the 3D object.

Note

This research is partially funded by the Spanish CICYT *DPI2002 – 02729* research project

References

1. Agarwal S., Roth D.: Learning a sparse representation for object detection. In the proceedings of the Seventh European Conference on Computer Vision, 2002.
2. Alvarado C., Oltmans M., Davis R.: A Framework for Multi-Domain Sketch Recognition. In the proc. of AAAI Spring Symposium on Sketch Understanding, 2002
3. Bandlow T., Hauck A., Einsele T., Farber G.: Recognising objects by their silhouette. In IMACS Conference on Comp. Engineering in Systems Applications (CESA'98), April. 1998
4. Borges D. L., Fisher R. B.: Class-Based Recognition of 3D Objects Represented by Volumetric Primitives. In R. Fisher, E. Trucco (eds.), British Machine Vision Conference 1996 (BMVC 96), (1)96:303-312, 1996.

5. Borotschnig H., Paletta L., Prantl M., Pinz A. : Appearance based active object recognition. *Image and Vision Computing*, (18): 715-727, 2000
6. Campbell R. J., Flynn P. J.: Eigenshapes for 3D object recognition in range data. In the proceeding of IEEE Conference C.V.P.R., Colorado, June 1999
7. Chen J.L., Stockman G. C.: 3D free form object recognition using indexing by contour features. *Computer Vision and Image Understanding*, 71, pp. 334-335, 1998
8. Cordella L. P., Foggia P., Sansone C., Vento M.: Learning structural shape descriptions from examples. *Pattern Recognition Letters*, 23, pp. 1427-1437, 2002
9. Davis R.: Position statement and overview: sketch recognition at MIT. In the proceedings of AAAI Spring Symposium on Sketch Understanding, 2002
10. Dudek G., Tsotsos J. K.: Shape representation and recognition from mutiscale curvature. *C. V. I. U.*, Vol. 68, N2, Nov. pp. 170-187, 1997
11. Goldmeier E.: Similarity in visually perceived forms. *Psycholog. issues*, Vol.8,1972
12. Hayward W. G.: Effects of outline shape in object recognition. *Journal of experimental psychology: human perception and Performance*, 24, pp. 427-440, 1998.
13. Hong D., Sarkodie-Gyan T., Campbell A. W., Yan Y.: A prototype indexing approach to 2-D object description and recognition. *Pattern Recognition*, Vol. 31,1998
14. Huang O. Y., Camps O. I., Kanungo T.: Object recognition using appearance-based parts and relations. *IEEE C.V.P.R. Conference*, Puerto Rico, 1997
15. Jaggi S., Karl W. C., Mallat S. G., Willsky A. S.: Silhouette recognition using high resolution pursuit. *Pattern Recognition*, Vol. 23, N5, pp. 753-771, May, 1999
16. Joshi T., Ponce J., Vijayakumar B., Kriegman D. J. : HOT curves for modeling and recognition of smooth curved 3D object. *IEEE C.V.P.R. Conference*, USA, 1994
17. Magnor M. A.: Geometry-based automatic object localization and 3D pose detection. In the proceeding of *IEEE S. S. I. A. I.*, Santa Fe, USA, pp. 144-147, 2002
18. Malaviya A., Peters L.: Fuzzy handwriting description language: FOHDEL. *Pattern Recognition*, Vol. 33, Issue 1, January 2000, pages 119-131
19. Meathrel R., Galton A.: Ahierarchy of boundary-based shape descriptors. In the proceeding of the seventeenth International Joint Conference on Artificial Intelligence, Vol. 2, pp. 1359-1364, Seattle, Washington, August, 2001
20. Medioni G. G., Franois A. R. J.: 3-D structures for generic object recognition. Invited conference paper, In the proceeding of the International Conference on Pattern Recognition, Vol. 1, pp. 30-37, Barcelona, Spain, Sept. 2000.
21. Mokhtarian F.: Silhouette-based isolated object recognition through curvature scale space. *IEEE PAMI*, 17, 1995
22. Murase H., Nayar S. K.: Visual learning and recognition of 3D objects from appearance. *International Journal of Computer Vision*, Vol. 14, pp. 5-24, 1995
23. Nelson R. C., Selinger A.: Improving appearance-based object recognition in cluttered backgrounds. In the proceeding of the 15th International Conference on Pattern Recognition (ICPR 2000), Vol1, Spain, Sept., 2000
24. Oliva A., Torralba A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, 44(1), 5-40, 2001
25. <http://oxygen.lcs.mit.edu/Overview.html>, 2002
26. Reinhold M., Deinzer F., Denzler J., Paulas D., Posl J.: Active appearance-based object recognition using viewpoint selection. In the proceeding of *Vision, Modeling, and Visualization Conference*, 2000, pp105-112, 2000, Berlin.
27. Sumi Y., Kawai Y., Yoshimi T., Tomita F.: 3D object recognition in cluttered environments by segment-based stereo vision. *I. J. C. V.*, 46(1), 2002
28. Super B. J.: Fast retrieval of isolated visual shapes. *C. V. I. U.*, 85, 1-21, 2002