Qualitative Representation of Planar Outlines

Richard Meathrel[1] and **Antony Galton**[1]

Abstract. A new boundary-based scheme for qualitatively representing planar outlines is described, consisting of a set of seventeen "atomic" tokens, and based on a combined discretisation of tangent bearing, curvature, and the rate of change of curvature. By grouping together strings of atomic tokens, higher-level primitive curve tokens can be specified (PCTs), that correspond to localised curve features of greater abstraction. We show how the primitives of existing boundary-based schemes may be defined as PCTs, and how associated token ordering graphs can be constructed that visually encode token-string syntax, based on the ordering constraints implicit in a set of PCT specifications. Because of the atomic nature of its building blocks, we propose that the scheme can be developed into a general framework for constructing sets of task-specific primitives, for use in application areas such as computer vision and qualitative spatial reasoning.

1 INTRODUCTION

Although there exist representations of shape in such areas of computer science as graphics, computer vision and pattern recognition, little attention has been given to the problem of *qualitatively* representing shape. In subdomains of AI such as qualitative spatial reasoning, symbolic schemes for describing shape are considered preferable to existing techniques, which are predominantly quantitative in kind. This is because, in reasoning at high levels (which we may regard as akin to "commonsense" reasoning), there is a need to deal with entities that are more abstract than those afforded by purely quantitative models.

In this paper, we focus our attention on the qualitative representation of planar curves, where we can distinguish between those approaches that are *boundary-based* and those that are *region-based*. Boundary-based schemes typically describe the type and placement of localised features round the bounding curve of a region (e.g. the *codon* primitives of [7][12], the process-based primitives of [9], the structural encodings of [3], and the qualitative curvature types of [5]). In contrast, region-based schemes base their descriptions on shape interior (e.g. axis-based techniques [1][2] and the RCC[2]-inspired approach of [4]). Here we concentrate on boundary-based schemes, in particular, those schemes which represent curves by strings of tokens that correspond to local features encountered whilst a curve is traversed; we call such representations *local feature schemes*. In this paper, we derive a set of *atomic curve tokens*, based on tangent bearing and curvature, which can be used to specify higher-level *primitive curve tokens*, in terms of which can be defined the primitives of existing local feature schemes.

[1] School of Engineering and Computer Science, University of Exeter, Exeter EX4 4PT, UK. E-mail: {R.C.Meathrel, A.P.Galton}@exeter.ac.uk

[2] The Region Connection Calculus, introduced in [10].

2 PLANAR OUTLINES

We restrict our attention in the current analysis to simple closed planar curves; we disregard here self-intersecting curves (e.g. lemniscates) and "pathological" curves (e.g. those with full or partial fractal structure, such as the Koch snowflake or a curve containing a stretch of a graph like $y = \sin(1/x)$ as it approaches the origin). Our convention for curve traversal is that we go round in the direction which keeps *figure* to the left and *ground* to the right, with an anticlockwise change in tangent bearing corresponding to positive curvature, and a clockwise change to negative curvature (see figure 1). When we go along a curve we may encounter points of tangent discontinuity, where the bearing of the tangent suddenly jumps from one value to another. We refer to such points as *kinks* in the curve, corresponding perceptually to angles and cusps.

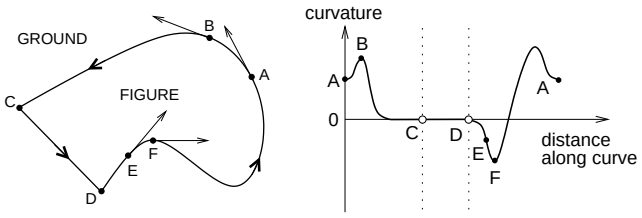


Figure 1. An example outline and its curvature plot. Points A and B, where the tangent is seen to rotate anticlockwise, indicate a segment of positive curvature. In contrast, E and F indicate negative curvature. The curve has two kinks, C and D, where the curvature is undefined.

2.1 Token-string descriptions

The defining characteristic of a local feature scheme is that it describes a curve by means of a string of tokens, $t_1 t_2 \ldots t_n$, that symbolise features of the curve that are considered significant. Token-strings for outlines are interpreted as cyclic, i.e. the feature represented by the last token in the string is followed, round the outline, by the feature represented by the first token. It is more accurate, therefore, to think of such token-strings as *rings*. Note that, although a given outline corresponds to a unique "token-ring", there will be (in general) a number of corresponding token-*strings*, e.g. an outline consisting of the four distinct features A, B, C, and D (in that order), is equally well described by any one of four strings: ABCD, BCDA, CDAB, and DABC. For the analysis that follows, we need not concern ourselves further with the issue of string correspondence[3].

[3] For a suitable method of deducing a single canonical token-string from a "token-ring" see [5, §3.7].

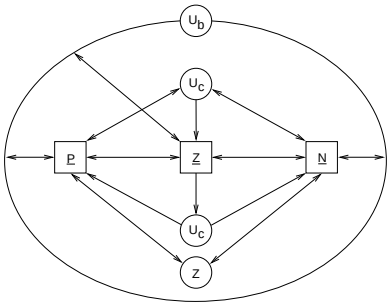
3 DERIVATION OF ATOMIC CURVE TOKENS

In this section we derive a set of fundamental "atomic" building blocks for qualitatively representing curves, such that the essential qualitative features of an outline's curvature plot (e.g. curvature minima/maxima, straight segments, points of inflection, points of undefined tangent-bearing/curvature) are associated with distinct tokens.

3.1 Base-level tokens

We start by deriving a set of *base-level* tokens, combining discretisations of tangent bearing and curvature. For curvature we use the discrete quantity space $\{+, 0, -, U\}$, since at any point along a curve the curvature may be positive, zero, negative, or undefined. For tangent bearing we use the space $\{D, U\}$: at kink points on a curve the tangent is undefined (U), at all other points it is defined (D). At any point p along a curve, then, we can think of the *curve state* as being specified by the pair $\langle b_p, c_p \rangle$, where b_p, c_p are the qualitative values of the tangent bearing and curvature at p, respectively. We obtain our set of base-level tokens by considering, for each valid pair, first whether the composite state can hold over an interval of points (allowing an interval interpretation) and second whether it can hold at a singular point (allowing a point interpretation)[4]. The table in figure 2 gives the set of six base-level tokens, $BaseTok$: there are three *interval tokens*, representing curve states that can persist over segments of curve (P, Z, and N) and three *point tokens*, representing curve states that can hold at singular points (Z, U_c, and U_b). Note that, by convention, interval tokens are underlined to distinguish them from point tokens.

	Tokens	
b c	*Interval*	*Point*
D $+$	P	
D 0	Z	Z
D $-$	N	
D U		U_c
U U		U_b



$BaseTok = \{\underline{P}, \underline{Z}, Z, \underline{N}, U_c, U_b\}$

Figure 2. The six tokens of $BaseTok$ and their ordering graph.

[4] Note that $\langle U, U \rangle$ is the only allowable pair with $b_p = U$, since whenever the tangent bearing is undefined at a point, the curvature must also be undefined (because it is the first derivative of the tangent bearing with respect to distance along the curve).

3.2 Token ordering graphs

The *token ordering graph* (TOG) for $BaseTok$ is also shown in figure 2. The nodes of interval tokens are shown as squares and the nodes of point tokens as circles. Conceptually, the purpose of a TOG is to provide a visual encoding of the ordering constraints for a particular set of tokens, by indicating which tokens can legitimately follow which other tokens in a string: an edge from X to Y indicates that XY is a legitimate substring. For our $BaseTok$ tokens, since each node represents an allowable curve state, an alternative interpretation is that an edge from X to Y indicates that, whilst traversing a curve, the qualitative state represented by Y may hold immediately after the qualitative state represented by X. By reference to the graph we can see, for example, that a point of undefined curvature (U_c) may follow an interval of positive curvature (P), and that the absence of an arrow from P to N indicates that it is not possible to have an interval of positive curvature *directly* followed by an interval of negative curvature[5]. There are two other points to note regarding the graph: (i) U_c maps to two nodes, thereby disallowing the token sequence $\underline{Z}U_c\underline{Z}$ which is unrealisable, and (ii) the outer ring of the diagram is not an edge, rather it is to be interpreted as a continuation of the node U_b itself.

By tracing a route through a TOG we can generate a valid token-string that, if treated as non-cyclic, defines an equivalence class of instantiable open curves. It is important to note, however, that when interpreted as cyclic such a string will not, in general, define a class of instantiable outlines, since additional constraints to ensure closure are required. The precise nature of the constraints required is an open question and beyond the scope of this paper.

As an example of representation using the tokens of $BaseTok$, consider the three shapes shown in figure 3 and their corresponding token-string descriptions:

Shape	*Description*
circle	P
square	$\underline{Z}\,U_b\,\underline{Z}\,U_b\,\underline{Z}\,U_b\,\underline{Z}\,U_b$
sausage	$\underline{P}\,Z\,\underline{N}\,Z$

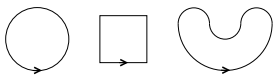


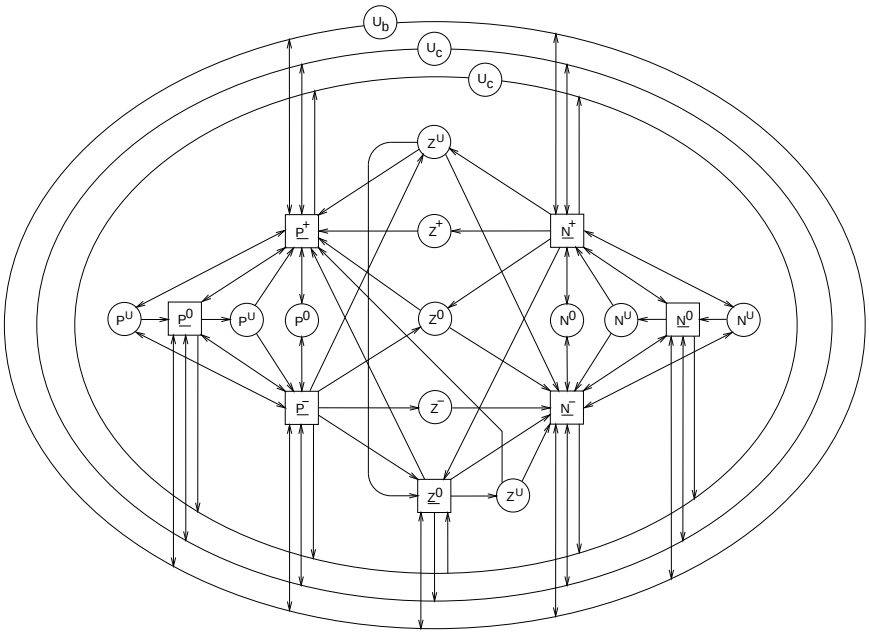
Figure 3. Describing a circle, a square, and a "sausage".

3.3 Atomic tokens

Although the base-level set of curve tokens allows us to describe all instances of the class of planar curves we are interested in representing, the discriminatory power it provides is limited. For this reason, we extend our set of $BaseTok$ tokens by supplementing the existing components of the curve state with the rate of change of curvature with distance along the curve (c'). In effect then, the curve state at any point p along a curve is now given by the triple $\langle b_p, c_p, c'_p \rangle$. As before, each composite state may be able to (i) persist over an interval of points, (ii) hold at a singular point, or (iii) both persist over an interval of points *and* hold at a singular point. The table on the left of figure 4 lists the fourteen valid composite combinations together with the seventeen tokens (state labels) we use to represent

[5] Instead, as we would expect, we have to go from P to N via either a point of undefined tangent bearing (U_b), a point of undefined curvature (U_c), a point of inflection (Z), or an interval of zero curvature (Z).

			Tokens	
b	c	c'	*Interval*	*Point*
D	+	+	$\underline{\mathsf{P}^{+}}$	
D	+	0	$\underline{\mathsf{P}^{0}}$	P^{0}
D	+	−	$\underline{\mathsf{P}^{-}}$	
D	+	U		P^{U}
D	0	+		Z^{+}
D	0	0	$\underline{\mathsf{Z}^{0}}$	Z^{0}
D	0	−		Z^{-}
D	0	U		Z^{U}
D	−	+	$\underline{\mathsf{N}^{+}}$	
D	−	0	$\underline{\mathsf{N}^{0}}$	N^{0}
D	−	−	$\underline{\mathsf{N}^{-}}$	
D	−	U		N^{U}
D	U	U		$\mathsf{U_c}$
U	U	U		$\mathsf{U_b}$



$$ATok = \{\underline{\mathsf{P}^{+}}, \underline{\mathsf{P}^{0}}, \mathsf{P}^{0}, \underline{\mathsf{P}^{-}}, \mathsf{P}^{\mathsf{U}}, \mathsf{Z}^{+}, \underline{\mathsf{Z}^{0}}, \mathsf{Z}^{0}, \mathsf{Z}^{-}, \mathsf{Z}^{\mathsf{U}}, \underline{\mathsf{N}^{+}}, \underline{\mathsf{N}^{0}}, \mathsf{N}^{0}, \underline{\mathsf{N}^{-}}, \mathsf{N}^{\mathsf{U}}, \mathsf{U_c}, \mathsf{U_b}\}$$

Figure 4. The seventeen tokens of $ATok$ and their ordering graph.

the allowable interval and point interpretations[6]. Collectively, these tokens make up the set of *atomic curve tokens* (or "atoms") that we refer to as $ATok$, the ordering graph of which is also shown in figure 4. With the elements of $ATok$ we are able to distinguish between curves which would be considered identical under $BaseTok$, since the atomic tokens afford a greater level of discriminatory power.

3.3.1 Proofs

We omit the proofs of correctness for the table and graph in figure 4 owing to space constraints. The table correctness is verified by rules determining the non-allowable interpretations (the empty cells), and exemplar curvature plots proving the realisability of the seventeen atoms. The graph is constructed from a pair of adjacency tables (one for *interval-interval* connections and one for *interval-point-interval* connections) and verified by additional rules and exemplar curvature plots. The proofs are available upon request.

4 PRIMITIVE CURVE TOKENS

Using the atoms of $ATok$ we can specify higher-level *primitive curve tokens* (PCTs) capable of representing local curve features that correspond to a succession of one or more curve states. The essential difference between atomic and primitive tokens is that an atomic token is a label representing a particular curve state, whereas a primitive token is a label representing a set of feature-defining *curve-state sequences*.

[6] Although there are $2 \times 4 \times 4 = 32$ possible combinations of b, c, and c', only the fourteen shown are valid because triples where $b = U \wedge (c \neq U \vee c' \neq U)$ or $b = D \wedge c = U \wedge c' \neq U$ are invalid.

4.1 Curve-state sequences

A *curve-state sequence* is defined as a valid string of atomic tokens that traces a route through $ATok$, with the consequence that curve-state sequences can be used to define local curve features. Simple features may be defined by strings of length one, e.g. $\underline{\mathsf{Z}^{0}}$ defines a straight line segment. More elaborate features are defined by strings of greater length. Given two sequences, $S_1 = a_1 a_2 \ldots a_x$ and $S_2 = b_1 b_2 \ldots b_y$, we say that:

- S_1 *may be immediately followed by* S_2 (written as "$S_1 \rightsquigarrow S_2$") if either (i) there exists a route $a_1 a_2 \ldots a_x b_1 b_2 \ldots b_y$ in $ATok$[7], or (ii) $a_x = b_1$ and there exists a route $a_1 a_2 \ldots a_x b_2 \ldots b_y$ in $ATok$ (so if both a_x and b_1 are interval tokens we are effectively merging them together, whereas if they are point tokens and are referring to the same actual point, we are collapsing them into one token occurrence).

4.2 PCT specifications

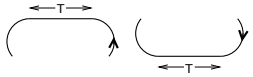
A PCT T is specified by a finite set of curve-state sequences, $T = \{S_1, \ldots, S_n\}$, such that:

- Each S_i is of the form $lc[id]tc$, where id is an *identity* substring and lc and tc are leading and trailing *context* substrings respectively. The identity substring serves to identify T with occurrences of id, while lc and tc specify the precise *context* in which T is to be so identified.

[7] Note that it's not enough just to require a direct connection from a_x to b_1, since it would not be correct to claim, for example, that $\mathsf{Z}^{\mathsf{U}}\underline{\mathsf{Z}^{0}}$ may immediately follow $\underline{\mathsf{Z}^{0}}$, because the sequence $\underline{\mathsf{Z}^{0}}\mathsf{Z}^{\mathsf{U}}\underline{\mathsf{Z}^{0}}$ is unrealisable.

- The type of T (whether it is an interval or point token) is determined by its identity substrings: if each id is a single point atom then T is a point token, whereas if each id contains an interval atom then T is an interval token.

As an example, consider the specification of a PCT that we wish to identify with straight lines that are "kinklessly" bounded by arcs of the same curvature sign:

$T = \{\ \underline{\mathsf{P}^0}\mathsf{U}_c[\underline{\mathsf{Z}^0}]\mathsf{U}_c\underline{\mathsf{P}^0},\ \underline{\mathsf{N}^0}\mathsf{U}_c[\underline{\mathsf{Z}^0}]\mathsf{U}_c\underline{\mathsf{N}^0}\ \}$ 

5 ORDERING GRAPH CONSTRUCTION

Given a set of PCTs, we can construct an associated TOG that encodes the ordering constraints implicit in the PCT specifications. A directed edge from a node X to a node Y indicates that, in a token-string, the substring XY is allowable. The task of constructing an ordering graph from a set of PCTs requires (i) a mapping between tokens and graph nodes to be found, and (ii) the connections between nodes to be determined.

5.1 Mapping tokens to graph nodes

We have already seen, in the base-level and atomic token TOGs, that it is sometimes necessary for tokens to map to more than one node. In the case of the $ATok$ TOG (figure 4), four of the point tokens (P^{U}, Z^{U}, N^{U}, and U_c) map to two graph nodes. In essence, more than one node will be required for a token T if there exist dependency relationships between the leading and trailing contexts of T. In the case of P^{U}, for example, if the preceding token (leading context) is $\underline{\mathsf{P}^0}$, then the succeeding token (trailing context) must not be $\underline{\mathsf{P}^0}$. For each of the tokens that map to a single node, there are no such dependencies: leading and trailing contexts are independent of one another. For TOGs constructed from base-level or atomic tokens, the leading and trailing contexts of a token are given by the curve states that can hold before and after that token. For TOGs constructed from primitive tokens, the contexts for a token are provided by the leading and trailing context-substrings of its curve-state sequences.

5.2 Connecting the nodes

In mapping a PCT to a set of nodes, we associate with each node a subset of the curve-state sequences of the PCT. The connection rule that determines whether a directed edge should be added between two nodes X and Y (not necessarily distinct), operates on the sequences associated with X and Y. Informally, we want to add a directed edge from X to Y if, along a curve, the feature represented by the sequences of Y may directly follow the feature represented by the sequences of X. More formally, then, given a set of graph nodes $\mathcal{N}$, we add a directed edge from a node N_x to a node N_y so long as the following node connection condition holds:

NCON: $(\exists S_i \in N_x, S_j \in N_y)(S_i \rightsquigarrow S_j)$

6 PCT ANALYSIS OF EXISTING LOCAL FEATURE SCHEMES

We may refer to a representation scheme for describing planar curves as PCT-based if its descriptions consist of strings of primitive curve tokens, i.e. if each curve feature it ascribes a token to can be built as a PCT from the atoms in $ATok$. In this section we show how the primitives of two well-known schemes from the literature can be specified and their associated ordering graphs constructed.

6.1 The process-based primitives of Leyton

Leyton [9] describes a scheme for modelling the changing shape of developing "natural forms" founded on a set of continuation and bifurcation processes. These processes are inferred from the presence of positive and negative curvature extrema round the contour of an object's silhouette. Shape descriptions consist of strings of symbols; there are five different symbols, one for zero-crossing points (0) and one for each of the four types of curvature extremum: M^+ and m^+ denote maxima and minima of positive curvature respectively, and M^- and m^- denote maxima and minima of negative curvature respectively. The PCT specifications for the five primitives are listed in figure 5. Each of the four extremum primitives are identified with stationary points that occur within a particular context, e.g. an M^+ is defined as a positive stationary point that is preceded by an interval of positive and increasing curvature ($\underline{\mathsf{P}^+}$), and followed by an interval of positive but decreasing curvature ($\underline{\mathsf{P}^-}$). The 0 primitive is identified with two points, Z^- and Z^+, since a zero-crossing may be from positive to negative *or* vice versa. For clarity, the two nodes for 0 in the TOG are annotated by ordinals that indicate the associated sequences. The graph edges (ordering constraints) are derived by applying NCON to each ordered pair of nodes.

$$
\begin{aligned}
M^+ &= \{\ \underline{\mathsf{P}^+}[\mathsf{P}^0]\underline{\mathsf{P}^-}\ \} \\
m^+ &= \{\ \underline{\mathsf{P}^-}[\mathsf{P}^0]\underline{\mathsf{P}^+}\ \} \\
M^- &= \{\ \underline{\mathsf{N}^+}[\mathsf{N}^0]\underline{\mathsf{N}^-}\ \} \\
m^- &= \{\ \underline{\mathsf{N}^-}[\mathsf{N}^0]\underline{\mathsf{N}^+}\ \} \\
0 &= \{\ \underline{\mathsf{P}^-}[\mathsf{Z}^-]\underline{\mathsf{N}^-},\ \underline{\mathsf{N}^+}[\mathsf{Z}^+]\underline{\mathsf{P}^+}\ \}
\end{aligned}
$$

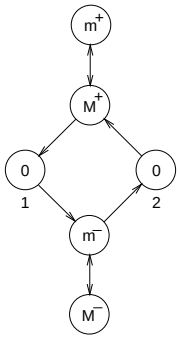


Figure 5. The PCT specifications and TOG for the primitives of Leyton.

6.2 The contour codons of Hoffman and Richards

Hoffman and Richards [7] present a set of *contour codons* for representing smooth planar curves for recognition purposes. Each codon is a segment of curve bounded by curvature minima that contains zero, one, or two points of zero curvature. Four codons are defined and symbolised by the tokens 0, 1^+, 1^-, and 2; contours of object silhouettes are described by strings of these tokens. We base our PCT specifications (see figure 6) on the set of codons given in [11] that includes a subdivision of the codon type 0 into 0^+ and 0^-, bringing the total number of codons to five[8]. The codon representation scheme is motivated by a theory regarding the perception of figure-ground reversal: the part-defining boundary points of a contour are postulated to be negative curvature minima, the proposed theory being that the reason why a curve looks different when figure and ground are reversed is that the change in the underlying codon description results in a different set of "parts" being perceived[9].

[8] They also include a straight-line type for completeness (labelled ∞, since it can be thought of as a curve segment with infinitely many points of zero curvature).

[9] See [8] for further discussion of part segmentation.

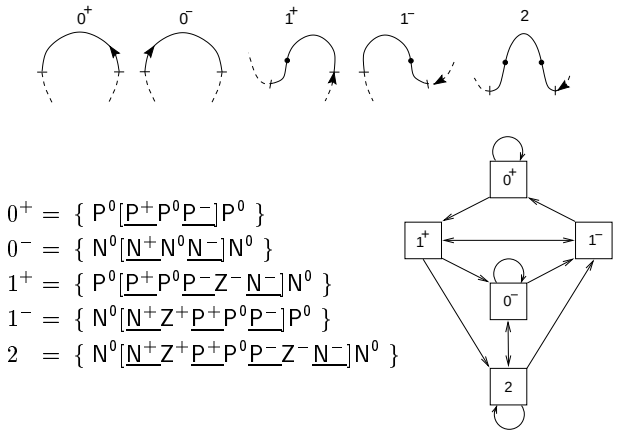


Figure 6. The PCT specifications and TOG for Hoffman and Richards' contour codons.

7 SUMMARY

We began by deriving a set of six base-level curve tokens from a composite discretisation of tangent bearing and curvature. In distinguishing just between points of defined and undefined tangent bearing, we ensured invariance with respect to rotation, translation, and uniform scaling. By supplementing the curve state with the rate of change of curvature, we were able to derive a set of qualitative states with significantly more discriminatory power: the seventeen atomic curve tokens of $ATok$. The atoms of $ATok$ can be used, in turn, to specify the semantics of PCTs: higher-level primitive curve tokens capable of representing more abstract local curve features. We outlined a procedure by which ordering graphs can be constructed that visually encode the ordering constraints implicit in a set of PCT specifications. Finally, we demonstrated that primitives of existing local feature schemes can be defined as PCTs, by providing PCT specifications and associated ordering graphs for two such schemes from the literature.

8 CONCLUSIONS AND FURTHER WORK

By combining discretisations of tangent bearing, curvature, and the rate of change of curvature, we can derive a set of "atomic" tokens that encode the important features of an outline's curvature plot, including kinks (points that correspond perceptually to angles and cusps). Such a set of tokens provides us with building blocks for qualitatively representing curves that are, in a sense, more fundamental than the primitives of existing local feature schemes, and also more general, since most schemes restrict their scope of representation to *kinkless* curves with continuously differentiable curvature plots.

As a consequence of their atomic quality, we can use the tokens of $ATok$ to specify the precise semantics of higher-level primitives (as demonstrated in §6) allowing us to highlight potential ambiguity in a primitive's definition. Taking Leyton's m^+ and M^- primitives as an example: each has two alternative specifications, depending on whether or not we allow the extrema to take the value zero[10]. Another ambiguity, regarding contour codons, is whether the specifications should include the delimiting minima in the leading/trailing context substrings or the identity substrings.

[10] We suspect that such points of zero curvature are not allowed, although Leyton does not seem to explicitly discount them.

We see the theory presented here as having two main purposes: (i) to serve as a theoretical basis for a unifying theory of boundary-based local feature schemes, and (ii) to serve as a point of departure from which a framework for constructing sets of task-specific primitives can be developed, for use in application areas such as computer vision and qualitative spatial reasoning. Ongoing work towards a unifying theory currently includes the PCT analysis of other schemes in the literature: the inclusion of cusps in Leyton's theory [6], Rosin's extended set of codons [12], and Galton and Meathrel's qualitative curvature types [5][11].

ACKNOWLEDGEMENTS

This research is supported by EPSRC award no. 97305965.

REFERENCES

[1] Michael Brady, 'Criteria for representations of shape', in *Human and machine vision*, eds., J. Beck, B. Hope, and A. Rosenfeld, 39–84, Academic Press, Inc., (1983).

[2] Jae-Moon Chung and Noboru Ohnishi, 'Chain of circles for matching and recognition of planar shapes', in *Proceedings of 15th IJCAI*, volume 2, pp. 1482–1487. Morgan Kaufmann Publishers, Inc., (1997).

[3] L. Cinque and L. Lombardi, 'Shape description and recognition by a multiresolution approach', *Image and Vision Computing*, **13**, 599–607, (1995).

[4] A. G. Cohn, 'A hierarchical representation of qualitative shape based on connection and convexity', in *Spatial Information Theory: A Theoretical Basis for GIS*, eds., Andrew U. Frank and Werner Kuhn, volume 988 of *Lecture Notes in Computer Science*, pp. 311–326. Springer-Verlag, (1995).

[5] Antony P. Galton and Richard C. Meathrel, 'Qualitative outline theory', in *Proceedings of 16th IJCAI*, ed., Thomas Dean, volume 2, pp. 1061–1066. Morgan Kaufmann Publishers, Inc., (August 1999).

[6] Patrick J. Hayes and Michael Leyton, 'Processes at discontinuities', in *Proceedings of 11th IJCAI*, volume 2, pp. 1267–1272. Morgan Kaufmann Publishers, Inc., (1989).

[7] D. D. Hoffman and W. A. Richards, 'Representing smooth plane curves for recognition: implications for figure-ground reversal', in *Proceedings of AAAI-82*, pp. 5–8. American Association for Artificial Intelligence, (1982).

[8] Donald. D. Hoffman and Manish Singh, 'Salience of visual parts', *Cognition*, **63**(1), 29–78, (1997).

[9] Michael Leyton, 'A process-grammar for shape', *Artificial Intelligence*, **34**, 213–247, (1988).

[10] D. A. Randell, Z. Cui, and A. G. Cohn, 'A spatial logic based on regions and connection', in *Proc. 3rd Int. Conf. on Knowledge Representation and Reasoning*, pp. 165–176, (1992). Cambridge MA, Oct. 1992.

[11] Whitman Richards and Donald D. Hoffman, 'Codon constraints on closed 2D shapes', in *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*, eds., Martin A. Fischler and Oscar Firschein, 700–708, Morgan Kaufmann Publishers, Inc., (1987).

[12] Paul L. Rosin, 'Multiscale representation and matching of curves using codons', *CVGIP: Graphical Models and Image Processing*, **55**(4), 286–310, (July 1993).

[11] Preliminary findings suggest that, for the qualitative curvature types, PCT specification will need to be extended in order to define $\supset$ and $\subset$, two primitives that correspond to a *variable* number of atoms.