Learning Complex Action Patterns with $\mathbf{CRG_{ST}}$

Walter F. Bischof and Terry Caelli

Department of Computing Science, University of Alberta, Edmonton, Alberta,
T6G 2E8, Canada, Email: (wfb,tcaelli)@ualberta.ca

Abstract. This paper deals with the problem of automatically compiling rules which describe complex actions in terms of the spatio-temporal attributes of labeled parts. Of particular interest is the exploration of a model-based approach to induction of part attributes constrained by known properties of the generation process. The resultant algorithm is based on constraint propagation over spatio-temporal decision trees which produces Horn clause descriptions which depict the spatio-temporal properties of parts and their relations which satisfy training conditions.

Keywords: relational learning, spatio-temporal patterns, action learning, domain knowledge constraints, conditional rule generation

1 Introduction

Most current techniques for the encoding and recognition of actions use numerical machine learning models which are not relational in the sense that they typically induce rules over numerical attributes which are not linked via an underlying data structure (e.g. a relational structure description). Therefore, these models assume that the correspondence between candidate and model features is known *before* rule generation (learning) or rule evaluation (matching) occurs. This assumption is dangerous when large models or large test data are involved, as is the case in complex actions involving, for example, the tracking of multiple limb segments of humans. On the other hand, well known symbolic relational learners like Inductive Logic Programming (ILP) are not efficient for numerical data. So, although they are suited to induction over relational structures (e.g. Horn clauses), they typically generalize or specialize over the symbolic variables and not so much over numerical attributes. Further, it is very rare that symbolic representation *explicitly* constrain the types of permissible numerical learning or generalizations obtained from training data.

Over the past six years we have explored methods for combining the strengths of both sources of model structures [1–3] by combining the expressiveness of ILP with the generalization models of numerical machine learning. We have produced a system for numerical relational learning which induces over numerical attributes in ways which are constrained by relational patterns. Our approach, Conditional Rule Generation (CRG), generates rules for the recognition of pattern fragments that are linked via an underlying relational structure.

Since it induces over a relational structure it requires general model assumptions, the most important being that the models are defined by a labeled graph where relational attributes are defined only with respect to specific vertices. These can be defined in a general way (e.g. they might be defined only for adjacent image regions), or they can be defined *explicitly* through model definitions. It is often the case that the properties of one part are physically controlled by others (as we will see for the case of human body motion where one limb segment controls the range of another). These models constrain the types of unary and binary features which can be used to resolve uncertainties (Figure 1).

In the following, we first describe briefly CRG [1] and then $\mathrm{CRG}_{\mathrm{ST}}$, a spatio-temporal extension of CRG. We discuss representational issues, model constraints, rule generation and rule application, and then illustrate our approach with several examples.

2 Conditional Rule Generation

In Conditional Rule Generation [1], classification rules for patterns or pattern fragments are generated that include structural pattern information to the

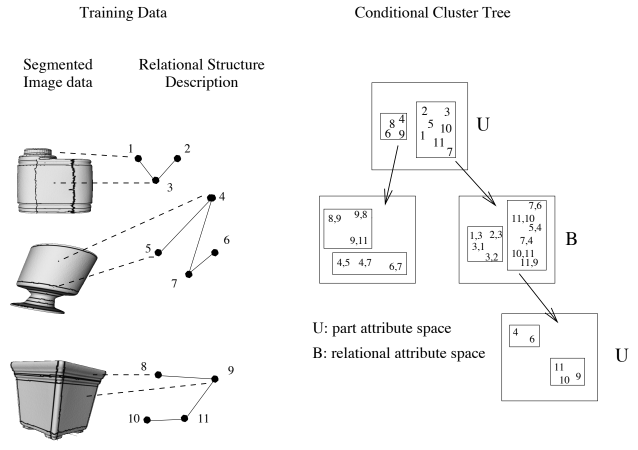

Fig. 1. Example of input data and conditional cluster tree generated by CRG method. The left panel shows segmented input data with a sketch of the relational structure descriptions generated for these data. The right panel shows a cluster tree generated for the data on the left. Classification rules are derived directly from this tree [5].

extent that is required for classifying correctly a set of training patterns. CRG analyzes unary and binary features of connected pattern components and creates a tree of hierarchically organized rules for classifying new patterns. Generation of a rule tree proceeds in the following manner (see Figure 1).

First, the unary features of all parts of all patterns are collected into a unary feature space U in which each point represents a single pattern part. The feature space U is partitioned into a number of clusters U_i. Some of these clusters may be unique with respect to class membership and provide a classification rule: If a pattern contains a part p_r whose unary features $\boldsymbol{u}(p_r)$ satisfy the bounds of a unique cluster U_i then the pattern can be assigned a unique classification. The non-unique clusters contain parts from multiple pattern classes and have to be analyzed further. For every part of a non-unique cluster we collect the binary features of this part with all adjacent parts in the pattern to form a (conditional) binary feature space UB_i. The binary feature space is clustered into a number of clusters UB_{ij}. Again, some clusters may be unique and provide a classification rule: If a pattern contains a part p_r whose unary features satisfy the bounds of cluster U_i, and there is an other part p_s, such that the binary features $\boldsymbol{b}(p_r, p_s)$ of the pair $\langle p_r, p_s \rangle$ satisfy the bounds of a unique cluster UB_{ij} then the pattern can be assigned a unique classification. For non-unique clusters, the unary features of the second part p_s are used to construct another unary feature space UBU_{ij} that is again clustered to produce clusters UBU_{ijk}. This expansion of the cluster tree continues until all classification rules are resolved or a maximum rule length has been reached.

If there remain unresolved rules at the end of the expansion procedure (which is normally the case), the generated rules are split into more discriminating rules using an entropy-based splitting procedure where the elements of a cluster are split along a feature dimension such that the normalized partition entropy $H_P(T) = (n_1 H(P_1) + n_2 H(P_2))/(n_1 + n_2)$ is minimized, where H is entropy. Rule splitting continues until all classification rules are unique or some termination criterion has been reached. This results in a tree of conditional feature spaces (Figure 1), and within each feature space, rules for cluster membership are developed in the form of a decision tree. Hence, CRG generates a tree of decision trees.

3 $\mathrm{CRG_{ST}}$

We now turn to $\mathrm{CRG_{ST}}$, the focus of this paper and a generalization of CRG from a purely spatial domain into a spatio-temporal domain. Data consist of time-indexed pattern descriptions, where pattern parts are described by unary features, part relations by (spatial) binary features, and changes of pattern parts by (temporal) binary features.

In contrast to other temporal learners like hidden Markov models [12] and recurrent neural networks [4], the temporal relations are not limited to first-order time differences but can involve more distant (lagged) temporal relations as a function of the data model and uncertainty resolution strategies. At the same time, CRG$_{\mathrm{ST}}$ allows for the generation of non-stationary rules, in contrast to stationary models like multivariate time series which also accommodate correlations beyond first-order time differences but do not allow for the use of different rules at different time periods.

3.1 Representation of Spatio-Temporal Patterns

A spatio-temporal pattern is defined by a set of labeled time-indexed attributed features, i.e. a pattern is defined as $P_i = \{p_{i1}(\boldsymbol{a} : t_{i1}), \ldots, p_{in}(\boldsymbol{a} : t_{in})\}$ where $p_{ij}(\boldsymbol{a} : t_{ij})$ corresponds to part j of pattern i with attributes $\boldsymbol{a}$ that are true at time j. The attributes $\boldsymbol{a}$ are defined with respect to specific labeled features, and consist of unary (i.e. single feature) attributes, spatial binary (i.e. spatial relational) and temporal binary (i.e. temporal relational) attributes, that is, $\boldsymbol{a} = \{\boldsymbol{u}, \boldsymbol{b}_s, \boldsymbol{b}_t\}$ (see Figure 2). Examples of unary attributes $\boldsymbol{u}$ include area, brightness, position; spatial binary attributes $\boldsymbol{b}_s$ include distance, relative size, and temporal binary attributes $\boldsymbol{b}_t$ include changes in unary attributes over time, such as size, orientation change, long range position change.

Our data model and consequently the rules generated are subject to several constraints, spatial and temporal adjacency (in the nearest neighbor sense) and temporal monotonicity, i.e. temporal indices for time must be monotonically increasing ("predictive" model) or decreasing ("causal" model). Further, we discuss additional constraints in Section 3.4, where induction over specific model-based relational structures is introduced. Although this limits the expressive power of our representation, it is still more general than strict first-order discrete time dynamical models such as hidden Markov models or Kalman filters.

For CRG$_{\mathrm{ST}}$ an "interpretation" then involves determining the smallest set of linked lists of attributed and labeled features, causally indexed (i.e. the starting times must be monotonically indexed) over time, which maximally index a given pattern, and it is defined by directed paths within the directed acyclic graph (DAG) which covers all examples and classes in the training set, as illustrated in Figure 2.

3.2 Rule Learning

CRG$_{\mathrm{ST}}$ generates classification rules for spatio-temporal patterns involving a small number of pattern parts subject to the following constraints: 1) The pattern fragments involve only pattern parts that are adjacent in space and time, 2) the pattern fragments involve only non-cyclic chains of parts, 3) temporal links are followed in the forward direction only to produce causal classification rules that can be used in classification and in prediction mode.

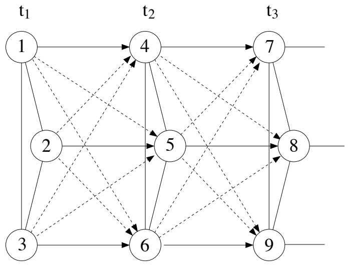

Fig. 2. Illustration of a spatio-temporal pattern consisting of three parts over three time-points. Undirected arcs indicate spatial binary connections, solid directed indicate temporal binary connections between the same part at different time-points, and dashed directed arcs indicate temporal binary connections between different parts at different time-points.

Rule learning proceeds in the following way: First, the unary features of all parts (of all patterns at all time points), $\boldsymbol{u}(p_{it})$, $i = 1, \ldots, n$, $t = 1, \ldots, T$, are collected into a unary feature space U in which each each point represents a single pattern part at any time point $t = 1, \ldots, T$. From this unary feature space, cluster tree expansion can proceed in two directions, in the spatial domain and in the temporal domain. In the spatial domain cluster tree generation proceeds exactly as described in Section 2 following spatial binary relations, etc. In the temporal domain, binary relations can be followed only in strictly forward (predictive) or backward (causal) directions, analyzing recursively temporal changes of either the same part, $\boldsymbol{b}_t(p_{it}, p_{it+1})$ (solid arrows in Figure 2), or of different pattern parts, $\boldsymbol{b}_t(p_{it}, p_{jt+1})$ (dashed arrows in Figure 2) at subsequent time-points. This leads to a conditional cluster tree as shown in Figure 1, except that the relational attribute spaces B can be either spatial or temporal, in accordance with the usual Minimum Description Length (MDL) criterion for Decision Trees[10].

3.3 Rule Application

A set of classification rules is applied to a spatio-temporal pattern in the following way. Starting from each pattern part (at any time point), all possible sequences (chains) of parts are generated using parallel, iterative deepening, subject to the constraints the only adjacent parts are involved and no loops are generated. Note, again, that spatio-temporal adjacency and temporal monotonicity constraints are used for rule generation. Each chain is classified using the classification rules. Expansion of each chain $S_i = < p_{i1}, p_{i2}, \ldots, p_{in} >$ terminates if one of the following conditions occurs: 1) the chain cannot be expanded without creating a cycle, 2) all rules instantiated by S_i are completely resolved, or 3) the binary features $\boldsymbol{b}_s(p_{ij}, p_{ij+1})$ or $\boldsymbol{b}_t(p_{ij}, p_{ij+1})$ do not satisfy the features bounds of any rule.

If a chain S cannot be expanded, the evidence vectors of all rules instantiated by S are averaged to obtain the evidence vector $\boldsymbol{E}(S)$ of the chain S. Further, the set $\mathcal{S}_p$ of all chains that start at p is used to obtain an initial evidence vector for part p:

$$\boldsymbol{E}(p) = \frac{1}{\#(\mathcal{S}_p)} \sum_{S \in \mathcal{S}_p} \boldsymbol{E}(S). \tag{1}$$

where $\#(\mathcal{S})$ denotes the cardinality of the set $\mathcal{S}$. Evidence combination based on (1) is adequate if it is known that a single pattern is to be recognized. However, if the test pattern consists of multiple patterns then this simple scheme can easily produce incorrect results because some some part chains may not be contained completely within a single pattern but “cross” spatio-temporal boundaries between patterns. This occurs when actions corresponding to different types cross can intersect in time and/or space. These chains are likely to be classified in a arbitrary way. To the extent that they can be detected and eliminated, the part classification based on (1) can be improved.

We use general heuristics for detecting rule instantiations involving parts belonging to different patterns. They are based on measuring the compatibility of part evidence vectors and chain evidence vectors. More formally, the compatibility measure can be characterized as follows. For a chain $S_i =< p_{i1}, p_{i2}, ..., p_{in} >$,

$$\boldsymbol{w}(S_i) = \frac{1}{n} \sum_{k=1}^{n} \boldsymbol{E}(p_{ik}) \tag{2}$$

where $\boldsymbol{E}(p_{ik})$ refers to the evidence vector of part p_{ik}. Initially, this can be found by averaging the evidence vectors of the chains which begin with part p_{ik}. Then the compatibility measure is used for updating the part evidence vectors using an iterative relaxation scheme [7]:

$$\boldsymbol{E}^{(t+1)}(p) = \Phi \left(\frac{1}{Z} \sum_{S \in S_p} \boldsymbol{w}^{(t)}(S) \otimes \boldsymbol{E}(S) \right), \tag{3}$$

where Φ is the logistic function, Z a normalizing factor $Z = \sum_{S \in S_p} w^{(t)}(S)$, and the binary operator $\otimes$ is defined as a component-wise vector multiplication $[a\ b]^T \otimes [c\ d]^T = [ac\ bc]^T$. The updated part evidence vectors then reflect the partitioning of the test pattern into distinct subparts.

3.4 Rule Generation using Domain Model Constraints

The definition of spatio-temporal patterns introduced in Section 3.1 is very general and applies to situations where no domain knowledge is available. Learning of patterns may be made more efficient through introduction of relational constraints based on domain knowledge. For example, for the recognition of human body movements, the spatial relation between hand and elbow may be much more diagnostic than the relation between hand and knee, or, more generally, intra-limb spatial relations are more diagnostic than inter-limb spatial relations. For these reasons, arbitrary model-based constraints can be introduced into the underlying relational structure, thus covering the range from fully-connected non-directed relational models to specific directed relational models. Obviously, in situations where no domain knowledge is available, the most general model should be used, and learning is consequently slower and sub-optimal. Conversely, when sufficient domain knowledge is available, strong constraints can be imposed on the relational model, and learning is consequently more efficient.

4 Example

The CRG$_{\mathrm{ST}}$ approach is illustrated in an example where the classification of four different variations of lifting movements were learned, two where a heavy object was lifted, and two where a light object was lifted. Both objects were either lifted with a knees bent and a straight back ("good lifting"), or with knees straight and the back bent ("bad lifting"). Thus there were four movement classes, 1) good lifting of heavy object, 2) good lifting of light object, 3) bad lifting of a heavy object, and 4) bad lifting of a light object. The movements are quite difficult to discriminate, even for human observers. This was done in order to test the limits of the movement learning system.

The movements were recorded using a Polhemus system [11] running at 120Hz for six sensors, located on the hip, above the knee, above the foot, on the upper arm, on the forearm, and on the hand of the left body side (see Figure 3). Each movement type was recorded five times. From the position data $(x(t), y(t), z(t))$ of these sensors, 3-D velocity $v(t)$ and acceleration $a(t)$ were extracted, both w.r.t. arc length $ds(t) = (dx^2(t) + dy^2(t) + dz^2(t))^{1/2}$, i.e. $v(t) = ds(t)/dt$ and $a(t) = d^2s(t)/dt^2$ [9]. Sample time-plots of these measurements are shown in Figure 4.

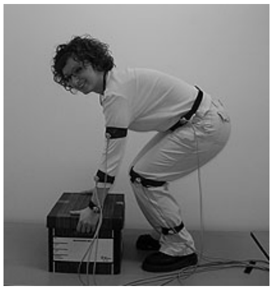

Fig. 3. Lifting a heavy object. The movement sensors were placed on the hip, above the knee, above the foot, on the upper arm, on the forearm, and on the hand of the left body side.

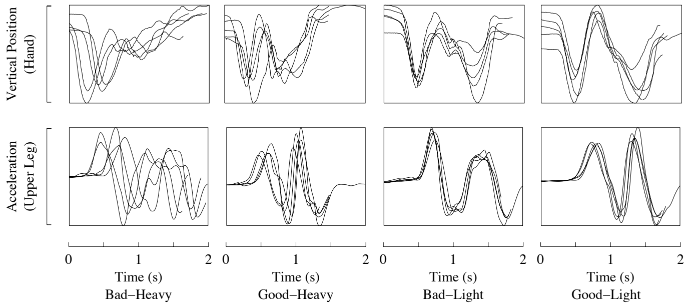

Fig. 4. Sample time-plots of the movement sequences illustrated in Figure 3. The first row shows time-plots for the vertical position of the sensor placed on the hand, the second row the acceleration of the sensor placed above the knee. The four columns show traces the four movement classes (see text for further details).

The spatio-temporal patterns were defined in the following way: At every time point, the patterns consisted of six parts, one for each sensor, each part being described by unary attributes $\boldsymbol{u} = [x, y, z, v, a]$. Binary attributes were defined by simple differences, i.e. the spatial attributes were defined as $\boldsymbol{b}_s(p_{it}, p_{jt}) = \boldsymbol{u}(p_{jt}) - \boldsymbol{u}(p_{it})$, and the temporal attributes were defined as $\boldsymbol{b}_t(p_{it}, p_{jt+1}) = \boldsymbol{u}(p_{jt+1}) - \boldsymbol{u}(p_{it})$.

Performance of CRG$_{\mathrm{ST}}$ was tested with a leave-one-out paradigm, i.e. in each test run, movement classes were learned using all but one sample, and the resulting rule system was used to classify the remaining pattern, as described in Section 3. The system was tested with three attribute combinations and four pattern models. The three attributes combinations were 1) $\boldsymbol{u} = [x, y, z]$, 2) $\boldsymbol{u} = [v, a]$ and 3) $\boldsymbol{u} = [x, y, z, v, a]$. The four pattern models were 1) a fully connected relational model (i.e. binary relations were defined between all six sensors), 2) a non-directional intra-limb model, i.e. binary relations were defined between hip - knee, knee - foot, upper arm - forearm, and forearm

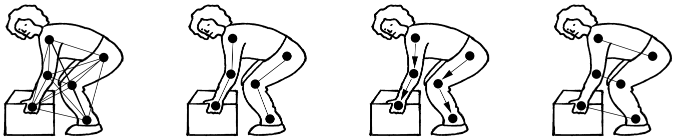

Fig. 5. Sketch of the four pattern models used for the recognition of lifting movements. From left to right, the sketches show the fully connected relational model, the non-directional intra-limb model, the directional intra-limb model and an inter-limb model. See text for further explanations.

Model	xyz	va	$xyzva$
fully connected	48.7 (85)	24.6 (30)	45.7 (75)
intra-limb non-directional	46.2 (75)	32.4 (32)	46.2 (75)
intra-limb directional	52.7 (85)	24.1 (20)	63.3 (90)
inter-limb non-directional	41.4 (60)	22.1 (5)	42.1 (70)

Table 1. Performance of $\mathrm{CRG}_{\mathrm{ST}}$ for learning four different types of lifting actions. The first column indicates what relational model was used, and the three remaining columns give average performance for three different attributes combinations (xyz = position in 3D; v = velocity: a = acceleration). Each cell gives raw percentage correct for a model + feature set combination. The number in parentheses gives classification performance under the assumption that a single movement pattern is present and is obtained from the former using a simple winner-take-all criterion.

- hand, 3) a directional intra-limb model (i.e. binary relations were defined as in 2) but only in one direction), and finally 4) an inter-limb model (i.e. binary relations were defined between hip - upper arm, knee - forearm, and foot - hand) (see Figure 5).

Results of these tests are shown in Table 1, for the attribute subsets and the pattern models just described. The results show that performance is fairly high, in spite of the fact that the movement patterns are not easy to discriminate for human observers. Best performance is reached for the intra-limb directional model (see Figure 5) and the full feature combination $xyzva$. Even though performance for feature combination va is very low, the two features improve, not unexpectedly, performance for the xyz feature combination [6].

An example of the rules which demonstrate their higher-order spatio-temporal nature is the following, with V = velocity; A = acceleration, ΔV = velocity difference between different sensors or for the same sensor over different time points, ΔA = acceleration difference between different sensors or for the same sensor over different time points:

if $U_i(t)$	any value
and $B_{ij}(t)$	$-57 \leq \Delta V \leq 114$ and $-580 \leq \Delta A \leq 550$
and $U_j(t)$	$A \leq 180$
and $T_j(t, t+1)$	$-249 \leq \Delta V \leq 73$ and $181 \leq \Delta A \leq 2210$
and $U_j(t+1)$	$17 \leq V \leq 24$ and $132 \leq A \leq 301$
then	this is part of a good-heavy lifting action

In plain language, rules like the one above read something like the following: If the relative velocity between the upper and lower limb is in the range [-57,114] and that of the relative acceleration in the range [-580,550], and the lower limb has an acceleration less than 180, and to the next time step, velocity change of the lower limb is in the range [-249,73] and that of acceleration change is in the range [181,2210], and at the next time point velocity of the lower limb is in the range [17,24] and that of acceleration in the range [132,301], then this is part of a good lifting of a heavy object.

5 Conclusions

In this paper, we have considered a new type of spatio-temporal relational learner which, like explanation-based learning [8], uses domain knowledge constraints to control induction over training data. The results show that such constraints can indeed improve performance of decision-tree type learners. There are still many open questions to be solved. Of particular relevance is the ability of the spatio-temporal learners to incorporate multi-scaled interval temporal logic constraints and how the spatio-temporal domain modeling can be further used to generate rules which are generated to be robust, reliable and permit discovery of new relations while, at the same time, render valid interpretations.

References

1. W. F. Bischof and T. Caelli. Learning structural descriptions of patterns: A new technique for conditional clustering and rule generation. *Pattern Recognition*, 27:1231–1248, 1994.
2. W. F. Bischof and T. Caelli. Scene understanding by rule evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:1284–1288, 1997.
3. T. Caelli and W. F. Bischof, editors. *Machine Learning and Image Interpretation*. Plenum, New York, NY, 1997.
4. T. Caelli, L. Guan, and W. Wen. Modularity in neural computing. *Proceedings of the IEEE*, 87:1497–1518, 1999.
5. T. Caelli, G. West, M. Robey, and E. Osman. A relational learning method for pattern and object recognition. *Image and Vision Computing*, 17:391–401, 1999.
6. J. Kittler, R. P. W. Duin, and M. Hatef. Combining classifiers. In *Proceedings of the International Conference on Pattern Recognition*, 1996.
7. B. McCane and T. Caelli. Fuzzy conditional rule generation for the learning and recognition of 3d objects from 2d images. In T. Caelli and W. F. Bischof, editors, *Machine Learning and Image Interpretation*, pages 17–66. Plenum, New York, NY, 1997.
8. T. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.
9. F. Mokhtarian. A theory of multiscale, torsion-based shape representation for space curves. *Computer Vision and Image Understanding*, 68:1–17, 1997.
10. J. R. Quinlan. MDL and categorical theories (continued). In *Proceedings of the 12th International Conference on Machine Learning*, pages 464–470, 1995.
11. F. H. Raab, E. B. Blood, T. O. Steiner, and H. R. Jones. Magnetic position and orientation tracking system. *IEEE Transactions on Aerospace and Electronic Systems*, AES-15:709–, 1979.
12. L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, New York, NY, 1993.