Sketched Symbol Recognition with a Latent-Dynamic Conditional Model

Vincenzo Deufemia, Michele Risi, Genoveffa Tortora
Dipartimento di Matematica e Informatica
Università di Salerno
84084 Fisciano(SA), Italy
{deufemia, mrisi, tortora}@unisa.it

Abstract—In this paper we present a recognizer of sketched symbols based on Latent-Dynamic Conditional Random Fields (LDCRF), a discriminative model for sequence classification. The LDCRF model classifies unsegmented sequences of strokes into domain symbols by taking into account contextual and temporal information. In particular, LDCRFs learn the extrinsic dynamics among strokes by modeling a continuous stream of symbol labels, and learn internal stroke sub-structure by using intermediate hidden states. The performance of our work is evaluated in the electric circuit domain.

Keywords—discriminative models; sketched symbol recognition; electric circuit diagrams.

I. INTRODUCTION

Hand-drawn diagrams are an essential means of communicating information in many domains, including electrical engineering, software engineering and web design [1], [2]. With the growing popularity of digital input devices there is increasing interest in building systems that can automatically interpret freehand drawings.

The recognition of symbols in hand drawn diagrams is particularly difficult since the symbols can be drawn by using a different stroke-order, -number, and -direction, even if the symbols are drawn by the same user. Sketch recognition includes complex activities such as clustering (i.e., to determine which strokes have to be grouped in order to represent a domain symbol) and segmentation (i.e., to split strokes if they contribute to more than one symbol). The lack of precision in hand-drawn diagrams leads to recognition ambiguities whose automatic resolution usually involves the analysis of the context of a symbol, i.e., other symbols close to the ambiguous symbol.

In this paper, we propose a discriminative classification scheme able to recognize sketched symbols in hand-drawn diagrams through the joint classification of the various parts they are composed of. In particular, we employ Latent-Dynamic Conditional Random Fields (LDCRFs) [3], a discriminative model introduced for visual gesture recognition. The model performs simultaneous stroke segmentation and labeling and is able to capture both intrinsic and extrinsic symbol dynamics. Indeed, the model incorporates hidden state variables which model the sub-structure of a symbol sequence and learn dynamics between symbol labels incorporating context into recognition of symbol parts. The association of a disjoint set of hidden states to each symbol label allows to perform efficient inference on LDCRF models by using belief propagation during training and testing.

In contrast to other discriminative classification models (e.g., neural networks and support vector machines) LDCRFs are able to model dependencies not only between input data and its labels, but also model dependencies between labels. This is particularly appealing in sketch recognition since we can exploit the joint classification task to incorporate context into recognition. Moreover, LDCRFs offer several advantages over previous CRF learning methods by explicitly modeling hidden state variables, which allow us to represent the latent sub-structure of symbols to be recognized.

In the following sections, we first introduce the proposed approach, then we describe the proposed discriminative model together with the training and inference procedures. Finally, we present experimental results in the electric circuit domain.

II. SKETCH RECOGNITION WITH A DISCRIMINATIVE MODEL

The proposed discriminative model for sketch recognition is able to analyze the sketch as a continuous sequence of points, and classifies a stroke segment by considering its intrinsic and extrinsic structure. In particular, the approach considers a sketch $S$ as a set of $k$ input points drawn on a two dimensional plane with an input device, where $S = \{p_1, \ldots, p_k\}$ with $p_i = (x_i, y_i) \in \mathbb{R}^2$. The sequence of points is partitioned into subsequences, named observations, through the application of an algorithm able to detect feature points in $S$. A feature point is computed considering line end points, high curvature points, overlapping points, and a threshold $t_{length}$ that limits the number of input points in an observation. The feature points are the representative elements of the observations. Fig. 1(a) shows a sketched voltage symbol and the detected feature points.

For each observation $z$ we compute a set of features $\theta_z$ used in the recognition process. In particular, we consider the following features: position, orientation, direction, length, and the sum of the distances between the points of the observation and the line that connects the first and the last
point of the observation. Such features describe the latent sub-structure of the stroke containing the observation.

The feature points extracted from a sequence of strokes can be considered as a graph, where the nodes represent feature points and the arcs link the nodes that are temporally and/or spatially neighbor. In particular, the features associated to each node are used to define two kinds of functions: *state functions* and *transition functions*. The first can be seen as a measure of how likely an observation will take a particular label given a sketch. In other words, it measures the compatibility between an observation \( z_i \) and its possible classifications, based on its features and the features of the observations that come first \( z_i \) in the sequence. The latter can be seen as a measure of how the labels at neighboring nodes should interact given the observed sketch. In other words, it measures the spatio-temporal consistency between observations by considering the spatial and temporal features associated to the observations. As an example, Fig. 1(b) (Fig. 1(c), resp.) shows the graph constructed from the state functions (transition functions, resp.) on the nodes of Fig. 1(a).

![Figure 1](image)

The graph obtained from state and transition functions represents a network of classifiers since each node employs its state and transition functions in order to associate to a stroke (or a part of it) a symbol label. The classifiers are trained in a discriminative way on sketch sentences providing a joint distribution over the labels conditioned on such sentences. Thus, the discriminative model employs such a network to infer a symbol label for each observation by exploiting contextual information (its neighbor observations in the graph) to solve ambiguity issues. Indeed, the decision made by one classifier influences the decisions of its neighbors.

Finally, the labels obtained from the inference process are grouped by using an agglomerative clustering algorithm [4]. In particular, we use the single linkage hierarchical clustering algorithm to merge the clusters having the smallest minimum pairwise distance between their labels. The distance between two labels \( w_i \) and \( w_j \) is defined as the spatial distance between the corresponding observations \( z_i \) and \( z_j \).

The clustering process starts by initializing each cluster with a symbol label. Then the algorithm iteratively merges the clusters having the minimum distance between them. The clustering process ends when the distances between each pair of clusters come through a threshold, which is computed as the mean of the distances between all labels.

Fig. 2 shows the graphical structure of the proposed recognition process. In this process, \( S \) is the sequence of points, \( z \) is an observation extracted from the sketch, \( w \) is the domain label (e.g., resistor, capacitor) assigned to \( z \), \( h \) represents a latent sub-structure variables, and \( L \) represents a label associated to a set of observations.

![Figure 2](image)

In the next section, we formally define the LDCRF discriminative model used in our approach.

III. LATENT-DYNAMIC CONDITIONAL RANDOM FIELDS

The LDCRF model [3] is a discriminative model for simultaneous sequence segmentation and labeling. It extends the traditional CRF [5] model by incorporating hidden state variables to model the sub-structure of input data and to capture both extrinsic dynamics and intrinsic structure.

Formally, in the context of sketch recognition, given a sequence of stroke observations \( z = \{ z_1, z_2, \ldots, z_m \} \), the LDCRF model predicts a sequence of class labels \( w = \{ w_1, w_2, \ldots, w_m \} \)
\{w_1, w_2, \ldots, w_m\}. Each \(w_j\) is a class label for the \(j\)-th stroke of a sequence and is a member of a set \(\mathcal{Y}\) of possible class labels, which corresponds to the domain symbols. For example \(\mathcal{Y} = \{\text{capacity, ground, resistor, voltage, wire}\}\) for the electric circuit domain. Each observation \(z_j\) is represented by a feature vector \(\phi(z_j)\). To model the substructure of the strokes, a vector of hidden variables \(h = \{h_1, h_2, \ldots, h_m\}\), not observable in the training examples, are incorporated in the model.

Given the above definitions, we define a latent conditional model as follows:

\[
P(w|z, \theta) = \sum_h P(w|h, z, \theta)P(h|z, \theta) \tag{1}
\]

where \(\theta\) are the parameters of the model.

In order to keep training and inference efficient, we restrict the model to have disjointed sets of hidden states associated with each class label. Each \(h_j\) is a member of a set \(\mathcal{H}_{w_j}\) of possible hidden states for the class label \(w_j\). We define \(\mathcal{H}\), the set of all possible hidden states to be the union of all \(\mathcal{H}_{w_j}\) sets. Since sequences will by definition have \(P(w|z, \theta) = 0\) for any \(h_j \notin \mathcal{H}_{w_j}\), otherwise 1, we can express our model as:

\[
P(w|z, \theta) = \sum_{h: \forall h_j \in \mathcal{H}_{w_j}} P(h|z, \theta) \tag{2}
\]

where \(P(h|z, \theta)\) is defined by using the usual conditional random field formulation:

\[
P(h|z, \theta) = \frac{1}{K_\theta(z, \mathcal{H})} \exp(\sum_j V_\theta(j, h_j, z) + \sum_j E_\theta(j, h_{j-1}, h_j, z)) \tag{3}
\]

where \(K_\theta(z, \mathcal{H})\) is the observation dependent normalization, which summarizes over all hidden state sequences in \(\mathcal{H}\):

\[
K_\theta(z, \mathcal{H}) = \sum_{h \in \mathcal{H}} \exp(\sum_j V_\theta(j, h_j, z) + \sum_j E_\theta(j, h_{j-1}, h_j, z)) \tag{4}
\]

\(\theta = \{\lambda_1, \lambda_2, \ldots, \mu_1, \mu_2, \ldots\}\) is the set of model parameters, \(V_\theta(j, h_j, z)\) and \(E_\theta(j, h_{j-1}, h_j, z)\) are feature functions on individual vertex \(j\) and edge \((j-1, j)\), respectively,

\[
V_\theta(j, h_j, z) = \sum_k \lambda_k s_k(j, h_j, z), \tag{5}
\]

\[
E_\theta(j, h_{j-1}, h_j, z) = \sum_k \mu_k t_k(j, h_{j-1}, h_j, z), \tag{6}
\]

where \(s_k\) are state functions that depend on a single hidden variable of the model, and \(t_k\) are transition functions that depend on pairs of hidden variables.

**A. Training LDCRFs**

Given a training set consisting of \(n\) labeled sequences \((z_i, w_i)\) for \(i = 1, \ldots, n\), training is performed by optimizing the objective function to learn the parameter \(\theta^*\):

\[
L(\theta) = \sum_{i=1}^n \log P(w_i|z_i, \theta) - \frac{1}{2\sigma^2} ||\theta||^2
\]

The first term of this equation is the conditional log-likelihood of the training data. The second term is the log of a Gaussian prior with variance \(\sigma^2\), i.e., \(P(\theta) \sim \exp\left(\frac{1}{2\sigma^2}||\theta||^2\right)\).

We use gradient ascent to search for the optimal parameter values, \(\theta^* = \arg \max_\theta L(\theta)\), under this criterion. The gradient of \(\log P(w_i|z_i, \theta)\) for one particular training sequence \((z_i, w_i)\) with respect to the parameters \(\lambda_k\) associated with a state function \(s_k\) can be written as:

\[
\sum_{a,j} P(h_j = a|w, z, \theta) s_k(j, a, z) -
\sum_{j,a,w'}\sum_{a,b} P(h_j = a, w'|z, \theta) s_k(j, a, z)
\]

which denote expectations with respect to the current model distribution, whereas the gradient of \(\log P(w_i|z_i, \theta)\) with respect to the parameters \(\mu_k\) associated with a transition function \(t_k\) can be written as:

\[
\sum_{a,b,j} P(h_j = a, h_k = b|w, z, \theta) t_k(j, a, b, z) -
\sum_{a,b,h,j} P(h_j = a, h_k = b, w'|z, \theta) t_k(j, a, b, z)
\]

**B. Feature Functions**

The state functions do not model the dependency of the entire observation sequence but, instead, depend only on a subset of observations preceding the current segment. In other words, \(s_k(j, h_j, z) = s_k(j, h_j, \tilde{z}_j)\), where \(\tilde{z}_j = [z_{j-m}, \ldots, z_j]\) is a window of size \(m + 1\) preceding the \(j\)-th segment. Assume \(z_j\) has dimension \(d\). We have \(|\mathcal{H}| \times d(m+1)\) state functions, and each corresponds to a pair \((h, l)\), where \(1 \leq l \leq d(m+1)\). The state function \(s_{hl}(j, h_j, z) = \delta(h, h_j) \tilde{z}_j(l)\), where \(\tilde{z}_j(l)\) is the \(l\)-th entry in \(\tilde{z}_j\) and \(\delta\) is the indicator function which takes the value 1 for hidden states with neighborhood relationship.

We have \(|\mathcal{H}| \times |\mathcal{H}|\) transition functions and each corresponds to a hidden variable pair \((h, h')\), denoted by \(t_{hh'}:\)

\[
t_{hh'}(j, h_i, h_k, z) = \delta(h_i, h_k) \delta(h', h_k) \quad \forall h_i, h_k \in \mathcal{H}
\]

The corresponding parameters \(\mu_{hh'}\) form an \(|\mathcal{H}| \times |\mathcal{H}|\) matrix that is essentially a transition matrix. It models both the external dynamics between strokes and the internal substructures of individual strokes [3].
C. Inference on LDCRFs

For testing, given a new test sequence \( z \), we want to estimate the most probable label sequence \( w^* \) that maximizes our conditional model:

\[
w^* = \arg \max_w P(w|z, \theta^*)
\]

where the parameter values \( \theta^* \) are learned from training examples.

Assuming each class label is associated with a disjoint set of hidden states, the previous equation can be rewritten as:

\[
w^* = \arg \max_w \sum_{h \in H_w} P(h|z, \theta^*)
\]

The likelihood measurement for a sketched segment \( s \) is equal to the marginal probability \( P(w_j = s|z, \theta^*) \). This probability is equal to the sum of the marginal probabilities for the hidden states part of the subset \( H_s \):

\[
P(w_j = s|z, \theta^*) = \sum_{h \in H_s} P(h|z, \theta^*)
\]

where \( z \) is the concatenation of all the feature vectors \( z_j \) for the entire sequence and \( \theta^* \) are the model parameters learned during training. The marginal probabilities are estimated by using the belief propagation algorithm.

IV. Experiments

To evaluate the proposed technique we implemented a prototype for electric circuit diagrams and experimented it with ten subjects having little experience in sketch-based interfaces. They took a lesson of 10 minutes to learn the principles of drawing by sketches and the main features of the prototype. The electric circuit symbols considered in the experiment are: capacitor, ground, resistor, voltage, and wire. The experiment is divided into two stages: training and recognition.

In the first stage the system was trained on the diagrams drawn by three subjects randomly selected. Thus, to train the LDCRF model we used 12 diagrams containing 60 non-wire symbols connected with 72 wire lines from which 871 observations were detected.

In the second stage we asked the other seven subjects to draw 2 circuit diagrams each containing 5 non-wire symbols. Thus, we gathered 21 circuit diagrams composed by 70 non-wire symbols connected with 62 wire lines.

The implemented prototype has been customized setting \( l_{\text{length}} = 50 \) points and by using an algorithm for computing angle curvatures that analyzes the angle obtained from three consecutive points.

We have measured the recognition performance of the system for each symbol of the domain language by determining the number of correctly identified symbols in each sketch. In particular, the recognition effectiveness has been measured by considering the precision and recall metrics [6].

These measures can assume values in the range \([0, 1]\). The confusion matrix in Table I shows the types of errors made by the proposed prototype. In particular, the rows contain the interpretations obtained for the drawn symbols, while the columns indicate the instances (mis)recognized for a given symbol.

Let us observe that three times the resistor symbol has not been recognized by our approach, this is due to the clustering algorithm that grouped the labels of the resistor together with the labels of the neighbor wires. The other misrecognitions are due to the shape similarity between the symbols, e.g., capacitor and ground.

Table I

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Cap</th>
<th>Ground</th>
<th>Resistor</th>
<th>Voltage</th>
<th>Wire</th>
<th>Unrecognized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacitor</td>
<td>19</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Ground</td>
<td>1</td>
<td>9</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0.9</td>
</tr>
<tr>
<td>Resistor</td>
<td>-</td>
<td>-</td>
<td>21</td>
<td>-</td>
<td>-</td>
<td>0.675</td>
</tr>
<tr>
<td>Voltage</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>14</td>
<td>-</td>
<td>1.0</td>
</tr>
<tr>
<td>Wire</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>59</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Precision 0.86 0.81 1.0 1.0 0.98
Recall 0.86 0.81 1.0 1.0 0.98

V. Conclusion

In this paper we have proposed a discriminative conditional model based on LDCRFs for recognizing hand-drawn diagrams. The model performs simultaneous stroke segmentation and labeling and is able to capture both intrinsic and extrinsic symbol dynamics. The experimental results have shown that the proposed model is able to achieve good results in terms of precision and recall.

REFERENCES


