Recognition of strings using nonstationary Markovian models: an application in ZIP code recognition

Bouchaffra, D.; Govindaraju, V.; Srihari, S.N.;
Dept. of Comput. Sci., State Univ. of New York, Buffalo, NY

This paper appears in: Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.
Issue Date : 1999
Volume : 2 - 179 Vol. 2
Meeting Date : 23 Jun 1999-25 Jun 1999
Print ISBN: 0-7695-0149-4
Cited by : 1
INSPEC Accession Number: 6346905
Digital Object Identifier : 10.1109/CVPR.1999.784626
Date of Current Version : 06 August 2002

ABSTRACT

This paper presents nonstationary Markovian models and their application to recognition of strings of tokens, such as ZIP codes in the US mailstream. Unlike traditional approaches where digits are simply recognized in isolation, the novelty of our approach lies in the manner in which recognitions scores along with domain specific knowledge about the frequency distribution of various combination of digits are all integrated into one unified model. The domain knowledge is derived from postal directory files. This data feeds into the models as n-grams statistics that are seamlessly integrated with recognition scores of digit images. We present the recognition accuracy (90%) achieved on a set of 20,000 ZIP codes

Recognition of Strings Using Nonstationary Markovian Models: An Application in ZIP Code Recognition

Djamel Bouchaffra, Venu Govindaraju, and Sargur Srihari

Center of Excellence for Document Analysis and Recognition (CEDAR)
Department of Computer Science (CEDAR)
State University of New York at Buffalo
Amherst, New York 14228–2567

Abstract

This paper presents Nonstationary Markovian Models and their application to recognition of strings of tokens, such as ZIP Codes in the US mailstream. Unlike traditional approaches where digits are simply recognized in isolation, the novelty of our approach lies in the manner in which recognitions scores along with domain specific knowledge about the frequency distribution of various combination of digits are all integrated into one unified model. The domain knowledge is derived from postal directory files. This data feeds into the models as *n-grams* statistics that are seamlessly integrated with recognition scores of digit images. A specially interesting facet of the model is its ability to excite and inhibit certain positions in the n-grams leading to Markov Random Fields.

We have applied the general theory developed for processing strings of any symbols to ZIP Codes. We present the recognition accuracy achieved on a set of 20,000 ZIP Codes. The performance is superior to the present system which ignores all contextual information and simply relies on the recognition scores of the digit recognizers.

1 Introduction

Researchers in all fields have shown the value of using contextual information to enhance the recognition of patterns. Speech recognition community was the first to model the contextual knowledge present in the English language [8]. However the signal and language information were used in two distinct but successive phases. It has been shown since that it is possible to integrate these bodies of information in a single phase [2, 3]. Besides, Shinghal and Toussaint [10] address the benefits of using transition probablities that are dependent on both the word position and length in recognition of machine-printed text. In both applications, the use of Markovian models is well justified, albeit, for different reasons. Speech recognition involves processing the sound patterns in real-time. This naturally imposes a temporal aspect on the problem. It is a well known fact in speech analysis that the human voice apparatus introduces the *co-articulation effect*, where the sound of a syllable is affected by the previous syllable uttered. In fact, most real world applications, where a stream of patterns is presented to a recognizer in a sequential manner, would lend themselves to Markovian models. At any instant of time, the recognition of the pattern in question is dependent on what immediately precedes it, known as the history of the event. The order of the Markovian process would dictate the extent of the history which has influence on the pattern at hand.

Unlike the temporal realm of speech recognition, the task of machine-printed text recognition does not have a natural temporal component. An entire word image is presented for recognition as a list of character images.

The single most important issue that is addressed in this paper is whether what is true with the English language (as in speech and word images) holds true with numeric strings. Let us consider all the ZIP Codes in the United States. If the source of all the ZIP Codes is Markovian, then once again, the identity of any numeric symbol at a given position will be confined to its neighborhood. This conjecture has not been verified either empirically or theoretically. Each numeral in a string is created in isolation, independent of the neighboring numerals in the string. For that matter, it would not be unusual for a writer to write the numerals in the string in an arbitrary order (write the last 2 digits and then the first 3 to its left).

Throughout this paper we draw from the application of ZIP Code recognition. We wish to emphasize at this point that there is no loss of generality in our approach. In fact, the method would be equally valid for printed words and speech and other numeric strings such as telephone numbers, social security numbers, form IDs *etc.*

Figure 1 describes the flow-down of the various processes involved. Starting with an image of a ZIP Code the very first step is to isolate 5 digit images by a process of segmentation. Next is the task of recognizing each digit image by a digit recognizer. The recognizer returns a ranked list of classes for each digit along with its confidence associated with each choice. This paper assumes the results of this stage and hence all that preceeds this stage is essentially a *black box.* No matter what the particular algorithm may be, digit recognizers invariably compute the "goodness" of match between the image and the symbolic representation of the digit. While the confidence measures returned by recognizers are adequate for most applications where recognition is the final stage of the application, there is a need for true probabilistic measures when the scores of the recognizer must be integrated with subsequent stages of recognition as is the case with postprocessing in ZIP Code recognition (Figure 1). We adopt a procedure called DPS (Derving Probability from Scores), which we have described in detail elsewhere [2], that simply maps the confidence values to probability values.

This leads us to the subject matter of this paper: *Postprocessing* by Nonstationary Markov Modeling. The input is the trellis of class choices with associated probabilities and the output of the postprocessing is a ranked list of ZIP Code candidates.

2 Background

The recognition of handwritten numeric strings is usually easier than alphabetic words in a given address. There are fewer digits than alphabets (10 *versus* 26) and digits touch each other less often. Present ZIP Code recognition procedures routinely perform the task of locating the ZIP Code within the address, followed by segmenting and recognizing the ZIP Code digits.

What makes the task of finding 5 isolated digits non-trivial is the occasional occurrence of adjacent digits touching each other. We will assume in this paper that the digits are isolated by a segmenter [9, 11] with a correct segmentation rate of 92.9%.

We use a Gradient Structural Concavity (GSC) [5] recognizer to recognize the isolated digits. The accuracy of the recognizer on isolated digits is 96% on a particular set of images.

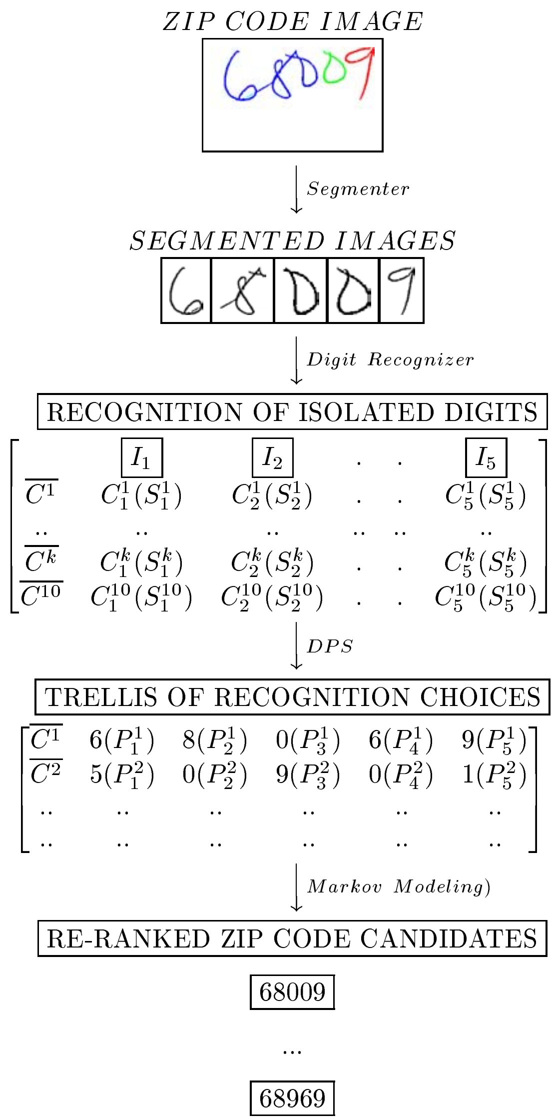

Figure 1: Starting with an image of a ZIP Code (numeric string of length $L = 5$), segmentation and digit recognition provide k (=1 . . . 10) class choices (C_t^k) for each isolated digit ($t \in \{1 \dots 5\}$). Each class choice C_t^k is associated with a score S_t^k. A procedure called DPS converts the scores into probability values P_t^k.*Postprocessing* by Markov Modeling, which is the focus of this paper,takes as input the trellis of class choices with associated probabilities to output a ranked list of ZIP Code candidates.

Let X_i represent the measurement made on an input pattern image I_i, where $\overline{I} = [I_1, I_2, \ldots I_L]$ is a sequence of subimages (*e.g.*, sequence of subimages of isolated digits). In some sense, the true image pattern is "hidden" from our modeling exercise and we are left to deal with the measurements alone. The classifier's k^{th} class choices are denoted by $\overline{C}^k = [C_1^k, C_2^k, \ldots C_L^k]$ based on the measurement vector $\overline{X}^k = [X_1^k, X_2^k, \ldots X_L^k]$ which corresponds to the subimages of $\overline{I}$. k ranges from 1 to 10 (the total number of classes) in the case of recognizing digits in a ZIP Code. When the pattern recognizer is treated as a blackbox, the measurement $\overline{X}$ is taken as the vector of confidence scores $\overline{S}$ returned by the recognizer rather than feature vectors of individual classes.

Considering all the class choices, we obtain a trellis $\mathcal{C} = \{\overline{C_t^k}\}(t \in \{1 \ldots L\},\ k \in \{1 \ldots c\})$ represented in Table 1. The parameter k is the choice level of the recognizer and c is the total number of classes. C_t^k is k^{th} recognition choice in column t of the *trellis*. Each class choice is associated with a score S_t^k returned by the digit (symbol) recognizer.

We denote the set of all class paths through the trellis by $\mathcal{P}$. Let's $\langle \overline{C}, \overline{X} \rangle$ be a particular class and its measurement path in $\mathcal{P}$. Our objective is to maximize $P(\overline{C} \mid \overline{I})$. We will drop $\overline{X}$ from our notations henceforth, as it is implied by reference to $\overline{C}$. The task at hand is simply to maximize the conditional probability $P(\overline{C} \mid \overline{S})$. By Bayes' rule, this would be equivalent to maximizing the quantity in Equation 1 with respect to $\overline{C}$.

$$\max_{\overline{C}:\ \overline{C} \in \mathcal{P}} P(\overline{C} \mid \overline{S}) = \max_{\overline{C}:\ \overline{C} \in \mathcal{P}} \left[\frac{P(\overline{S} \mid \overline{C}) \times P(\overline{C})}{P(\overline{S})} \right]. \quad (1)$$

Given that $P(\overline{S})$ is independent of $\overline{C}$, the task at hand is one of maximizing $P(\overline{S} \mid \overline{C}) \times P(\overline{C})$ over all possible paths $\overline{C}$ in the trellis. For a vector of length L, this amounts to considering a total of 10^L possibilities.

The postprocessing model has the capability of promoting candidates with lower scores to the top. An obvious extension to this phenomenon suggests that those digit choices that are not even on the list of recognition class choices returned by a recognizer, do not have a chance of moving to the top. Typically, digit recognizers return just the top 2 or 3 recognition choices. Let us say the maximum is 3. This leads to 3^5 ZIP Code candidates that can be generated by the 3×5 *trellis*.

Maximizing the expression $P(\overline{S} \mid \overline{C}) \times P(\overline{C})$ over $\overline{C}$ would be equivalent to maximizing any monotonically increasing function of the same expression. Hence, our objective could be re-stated as

$$\max_{\overline{C}:\ \overline{C} \in \mathcal{P}} \left[log\ P(\overline{S} \mid \overline{C}) \ + \ log\ P(\overline{C}) \right]. \quad (2)$$

	$\boxed{I_1}$	$\boxed{I_2}$	$\boxed{I_3}$	..	$\boxed{I_L}$
$\overline{C^1}$	$\langle C_1^1, S_1^1 \rangle$	$\langle C_2^1, S_2^1 \rangle$	$\langle C_3^1, S_3^1 \rangle$	..	$\langle C_L^1, S_L^1 \rangle$
$\overline{C^2}$	$\langle C_1^2, S_1^2 \rangle$	$\langle C_2^2, S_2^2 \rangle$	$\langle C_3^2, S_3^2 \rangle$	..	$\langle C_L^2, S_L^2 \rangle$
$\overline{C^3}$	$\langle C_1^3, S_1^3 \rangle$	$\langle C_2^3, S_2^3 \rangle$	$\langle C_3^3, S_3^3 \rangle$	..	$\langle C_L^3, S_L^3 \rangle$
..	..	..	..	..	..
$\overline{C^c}$	$\langle C_1^c, S_1^c \rangle$	$\langle C_2^c, S_2^c \rangle$	$\langle C_3^c, S_3^c \rangle$	..	$\langle C_L^c, S_L^c \rangle$

Table 1: 2D-Trellis representation of class candidates with their scores.

For sake of brevity, we will denote a path as $(\overline{C}, \overline{S}) = (C_t, S_t)$ for $(t = 1...L)$ composed of a class C_t and its measurement (score) vector S_t. Using the assumption of class conditional independence of scores and the fact that each score S_i depends only on class C_i, the simplification of Equation 2 is presented in Equation 3.

Each of the two terms in the right hand side of Equation 3 bear diverse bodies of knowledge. The first term, $P(S_t \mid C_t)$ is the probability of a measurement of a class at a particular node in the path $\overline{C}$ given the class C_t at this node. Since each node (C_t, S_t) in a path of the trellis $\mathcal{C}$ is a pair (C_t^k, S_t^k) where $(k = 1, ..., c)$ is the class choice and $(t = 1, ..., L)$ the position of this class in the path, the maximization problem of Equation 3 involves conditional probabilities $P(S_t^k \mid C_t^k)$. We assume that this conditional probability is independent of time (or position): $(P(S_t^k \mid C_t^k) = P(S^k | C^k))$[1]. This latter quantity is the probability of a measurement returned by a digit recognizer given the class of the image. Computing this probability is non-trivial. Indeed, computing its Bayes derivative, $P(C^k | S^k)$ is equally difficult. We have described a methodology of computing this probability elsewhere [2], in sufficient detail. For subsequent integration with the postprocessing module, it is crucial that the scores be converted to probability values (P_t^k) so that the Bayesian framework can be adopted.

We use the nonstationary property of a Markov chain in the language part.

$$log\ P(C_1) + \sum_{t=2}^{t=L} log\ P(C_t \mid C_1, C_2, ..., C_{t-1})$$

[1] This assumption is not always true since digit images at either end of a string contain less ligature and thus may impact the recognition scores.

$$\max_{C_t}\left[\sum_{t=1}^{t=L}(\ log\ P(S_t \mid C_t))\ +\ \left(log\ P(C_1)+\sum_{t=2}^{t=L}\ log\ P(C_t \mid C_1, C_2, ..., C_{t-1})\right)\right]. \quad (3)$$

by highlighting the temporal aspect of the class distribution. In fact, the position of a same class (or digit) is a discriminative factor in a ZIP Code ($P(C^k_{t_i}) \neq P(C^k_{t_j})$ if $i \neq j$).

These facts allow the simplification of Equation 3 to Equation 4. Equation 4 shows that the language part is essentially a combination of n-gram probabilities, where $1 \leq n \leq L$.

3 Objective

The focus of this paper resides in the computation of the second term, $P(\overline{C})$ (Equation 3). In some sense, this term captures the *prior* contextual knowledge that we have set out to model. This will be dealt with in section 5.

Our objective can be simply stated as developing a postprocessing model that produces the maximally accurate recognition of a string of symbols given a *trellis* of symbols and their associated scores by a recognizer. In the case of ZIP Code images, a succint description of the problem is illustrated by Figure 2.

4 Previous Work

Recognition of handwritten numeric strings has been tackled by several researchers in the literature. Notably, [11] describes a "segment and then recognize" approach for the application of postal ZIP Code recognition. While there are applications of numeric string recognition other than ZIP Codes, such as recognizing courtesey amounts on bank checks [7], the ZIP codes have the distinguishing feature of having a fixed length. Much of the positional (temporal) aspect of our model depends on this very fact. Choosing the right model and of the right order has been discussed in the context of bank check recognition [7]. However, we believe that a systematic study as presented in this paper has been lacking in the literature.

5 Modeling Contextual Knowledge

For the application of ZIP Code recognition, the appropriate model that represents the "structure" of the contextual knowledge must be determined. There are two ways of approaching this task. First, we experimentally test all possible models (in terms of degrees of Markovian processes) and all their combinations as an empirical way of determining the most suitable model. A second approach would be one where we attempt to understand the structure of the ZIP code, how different digits relate to one another, and derive a model that matches this understanding. Of course, it would be intutively satisfying if both approaches point to the same model. In the following section we explore both scenarios.

5.1 Empirical Evaluation of Models

If the numeric strings are all of a fixed length, (L in the discussion above) then the contextual information can be further qualified. By making the probabilities specific to their position within the string, one can better exploit the context. This adaptation is referred to in the literature as a *non-stationary* Markov Model [1]. The context that we are modeling is the probability, $P(C_t = \alpha)$ of a particular class, α, in a particular position, t in the numerical string, *given* the occurrence of a particular class (*say*, β) in the *previous* position, $t-1$: $P(C_t = \alpha) \mid C_{t-1} = \beta)$. Note that the choice of the variable t for denoting the position in the string is intentional in that it is the counterpart of the time parameter in temporal Markov models[2].

This notion of *prior* class conditionals ($P(C_t = \alpha \mid C_{t-1} = \beta)$) that we describe is similar to that of positional *n-grams* described in the literature [10].

- *Markovian Model of 0^{th} order:* It represents the case where there is no contextual information shared between neighboring numerals in a string.

  In the ZIP Code application, there are $\binom{5}{1}$ possible position 1-grams. This is represented by 5 column vectors, one for each position (t) of the ZIP Code. Each entry in the column vector, $U_t[i]_{i\in\{0...9\}}$, represents the *prior* probability $P(C_t = \alpha)$, the frequency of occurrence of digit α in position $t_{1...5}$.

- *Markovian Model of 1^{st} order:* Conditional probabilities of the first order are represented by 2-dimensional matrices $B_{t_1,t_2}[i,j]$ where $i,j \in \{0\ldots 9\}$; $t_1, t_2 \in \{1\ldots 5\}$, $t_1 \neq t_2$. It is to be noted that t_1 and t_2 need not be adjacent, *i.e.*, t_2 is not necessarily equal to $t_1 \bigoplus 1$. Furthermore,

[2] In order to avoid defining special cases for the boundary conditions, we will assume the numerical string to be circular. *Modulo L* addition (denoted henceforth by the operator $\bigoplus$) will imply that the successive neighbor of the digit in the last (L^{th}) position is the first digit ($L \bigoplus 1 = 1$).

$$\max_{C_t^{k_t}} \left[\sum_{t=1}^{t=L} log \; \frac{P(C^{k_t} \mid S^{k_t})}{P(C^{k_t})} \; + \; log \; P(C_1^{k_1}) + \sum_{t=2}^{t=L} log \; P(C_t^{k_t} \mid C_1^{k_1}, C_2^{k_2}, ..., C_{t-1}^{k_{t-1}}) \right]. \quad (4)$$

$$\longrightarrow \boxed{I_1}\boxed{I_2}\boxed{I_3}\boxed{I_4}\boxed{I_5} \longrightarrow \begin{bmatrix} 6(0.9) & 8(0.5) & 0(0.7) & 9(0.2) & 9(0.3) \\ 4(0.7) & 0(0.4) & 9(0.5) & 0(0.1) & 1(0.2) \\ .. & .. & .. & .. & .. \end{bmatrix} \longrightarrow \boxed{68009}$$

Figure 2: I/O behavior of the postprocessing module. The input is a *trellis* of classifier choices assoicated with probability values. The ouput is the best ZIP Code candidate based on the probabilities of recognition as well as the Markovian modeling as described in this paper.

we will distinguish between the ordering of t_1 and t_2. Thus the entry $B_{t_1,t_2}[i,j] \neq B_{t_2,t_1}[i,j]$.

In the ZIP Code application there are $2 \times \binom{5}{2}$ possible position pairs of t_1 and t_2. This would result in 20 matrices of size 10 $\times$ 10 each[3].

Not all combinations of t_1 and t_2 are contiguous. In fact, of the 20 combinations, only 8 of the combinations are contiguous in Relation 5.

- *Markovian Model of 2^{nd} order:* Conditional probabilities of the second order are represented by 3-dimensional matrices $T_{t_1,t_2,t_3}[i,j,k]$ where $i,j,k \in \{0 \ldots 9\}$, $t_1, t_2, t_3 \in \{1 \ldots 5\}$, $t_1 \neq t_2 \neq t_3$.

  In the ZIP Code application, there are $2 \times \binom{5}{3}$ possible position triples of of t_1, t_2, and t_3. This would result in 40 matrices of size 10 $\times$ 10 $\times$ 10 each.

- *Markovian Model of 3^{rd} order:* Conditional probabilities of the third order are represented by 4-dimensional matrices $Q_{t_1,t_2,t_3,t_4}[i,j,k,l]$ where $i,j,k,l \in \{0 \ldots 9\}$, $t_1, t_2, t_3, t_4 \in \{1 \ldots 5\}$, $t_1 \neq t_2 \neq t_3 \neq t_4$.

  In the ZIP Code application there are $2 \times \binom{5}{4}$ possible position quadrapules. This would result in 10 matrices of size 10 $\times$ 10 $\times$ 10 $\times$ 10 each.

- 5-grams information would correspond to a Markovian Model of the 4^{th} order. In the ZIP Code application there is just one $(\binom{5}{5})$ possible position 5-gram. The corresponding matrix $P_{t_1,t_2,t_3,t_4,t_5}[i,j,k,l]$ where $i,j,k,l \in \{0 \ldots 9\}$, $t_1, t_2, t_3, t_4, t_5 \in \{1 \ldots 5\}$, $t_1 \neq t_2 \neq t_3 \neq t_4 \neq t_5$ would be a 5 dimensional matrix.

[3]Traditional Bigram tables would simply note the frequency of occurrence of each digit pair.

5.2 Markov Random Fields

Relation 5 shows a phenomenon that we call *triggering*. The assignment of a digit at a particular position *triggers* the assignment of a digit at any position to the right (front) of it, keeping in mind that the positions are in a circular list. The concept of triggering is more general than that of *chaining* where the triggers are always in adjacent positions. It is easy to see that the ordering of the positions can be changed so that the non-contiguous combinations become contiguous (and *vice versa*) without altering the problem at hand. Triggering allows us to model the correlation between digits at any two positions in the string. The information content of the non-contiguous combinations could be equally if not more informative. Unlike the case of text strings, numeric strings usually do not carry any special meaning in adjacent groups of digits. It is for the experiments to determine the true underlying structure. In essence what we have uncovered here is what has long been understood as the Markov Random Fields (MRF). The states are in no particular order other than being correlated. Hence the concept of triggering that does not strictly follow the Markov chain is captured adequately by MRF [4].

If ζ is the MRF over Γ describing the class distribution in a ZIP Code, then the condition upon ζ is for every t=1,...,L.

$$P(\zeta_t = C_t | \zeta_u = C_u, t \neq u) = P(\zeta_t = C_t | \zeta_u = C_u, u \in \Gamma_t).$$

We use Γ_C (for class C) to define the neighborhood

$$
\begin{bmatrix} B_{t_1,t_2}[0,0] & . & . & . & B_{t_1,t_2}[0,9] \\ . & . & . & . & . \\ . & . & . & . & . \\ B_{t_1,t_2}[9,0] & . & . & . & B_{t_1,t_2}[9,9] \end{bmatrix} = \begin{bmatrix} P(C_{t_1}=0 \mid C_{t_2}=0) & . & . & . & P(C_{t_1}=0 \mid C_{t_2}=9) \\ . & . & . & . & . \\ . & . & . & . & . \\ P(C_{t_1}=9 \mid C_{t_2}=0) & . & . & . & P(C_{t_1}=9 \mid C_{t_2}=9) \end{bmatrix} \tag{5}
$$

$$
(t_1, t_2) \in \{1 \ldots 5\}^2 \, ; t_1 \neq t_2
$$

assigned to a MRF class distribution. The MRF for position shown by $\boxtimes$ in the string is shown by $\boxplus$. $\forall t \geq L, t = t \bigoplus L$.

$$
\begin{cases} \Gamma_1 = \{3\} \\ [\boxtimes \;\; \square \;\; \boxplus \;\; \square \;\; \square] \\ \Gamma_2 = \{4\} \\ [\square \;\; \boxtimes \;\; \square \;\; \boxplus \;\; \square] \\ \Gamma_t = \{t-2, t+2\} \quad 3 \leq t \leq L \\ \begin{bmatrix} \boxplus & \square & \boxtimes & \square & \boxplus \\ \boxplus & \boxplus & \square & \boxtimes & \square \\ \boxplus & \square & \boxplus & \square & \boxtimes \end{bmatrix} \end{cases} , \tag{6}
$$

5.3 Modeling the Structure of ZIP Codes

We have collected the n-gram tables (matrices) from the US Postal Delivery Point File. The ZIP Code consists of five digits. The first digit designates a broad geographical area of the United States, ranging from 0 for the Northeast to 9 for the West. The following two digits pinpoint population concentrations and reflect Sectional Center Facilities (SCF) that act as hubs in the transportation networks, (see Table 2). The final two digits designate small post offices. There are about 43,000 ZIP Codes in the United States that are valid. Only 43% of all possible 5 digit strings are valid. Of these, some ZIP Codes are more common in the mailstream than others. Surely, the volume of mail destined for Manhattan, NY is far greater than the volume destined for Boise, ID. This information is used to weigh the frequency of occurrence of different digits in different positions of the ZIP Code.

An intuitive reading of the structure of the ZIP Code would suggest that the 2^{nd} and 3^{rd} digits of the ZIP Code are related to the 1^{st} and the 4^{th} and 5^{th} digits depend on the first 3 digits (Relation 7).

Let us assume that the recognition choice of the ZIP Code under consideration is: $[u\ v\ w\ x\ y]$, *i.e.*, $[I_1 \equiv u,\ I_2 \equiv v,\ I_3 \equiv w,\ I_4 \equiv x,\ I_5 \equiv y]$. Then the objective is simply one of evaluating the probability of the string $u\ v\ w\ x\ y$. Equation 7 is what one would expect to be the probability of a sequence of 5 symbols. The probability of the ZIP Code $[u\ v\ w\ x\ y]$ is given by Equation 8 using terms from all the 5 Markovian orders.

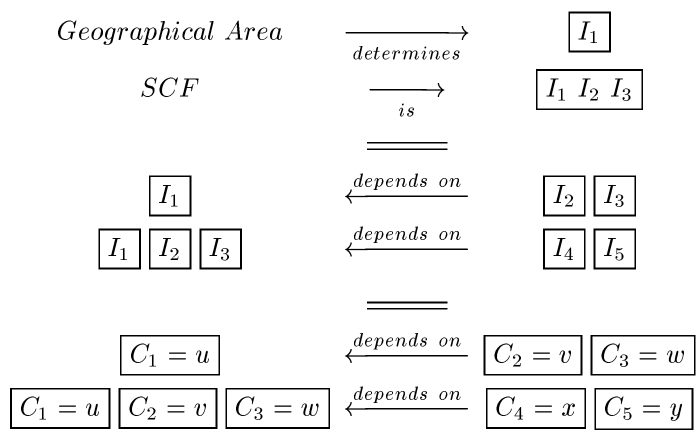

Table 2: Structure of a 5 digit ZIP Code based on its design. The structure is enriched by the MRF modeling.

6 Experiments

We have experimented with various combinations of *triggers* and different orders of the Markov modeling. Not surprisingly, the best results were obtained for the modeling based on Equation 8 which is a derivative of Equation 7. This indeed is intutively satisfying that the model which reflects our understanding of the correlation of digits in the ZIP code does perform the best. ZPP (ZIP Code Post Processing) software was tested on 4 different sets of ZIP Code images that were automatically extracted by the Handwritten Address Interpretation system [6]. GSC refers to the ZIP Code recognizer that is simply based on the recognition confidences of the GSC digit recognizer [5].

Table 3 shows a consistent 2% improvement on GSC. In fact, if the top 2 choice candidates are considered, the improvement margin is at 4%.

7 Summary

We have presented a nonstationary Markov model that merges images (signal) and context (language) in a fully Bayesian framework. The results obtained are promising. The Delivery Point file of the US Postal Service is used to generate a list of every valid ZIP Code paired with the number of records in the ZIP

$$P(C_1 = u,\ C_2 = v,\ C_3 = w,\ C_4 = x,\ C_5 = y) = \\ P(C_1 = u)\ \times \\ P(C_2 = v,\ C_3 = w \mid C_1 = u)\ \times \\ P(C_4 = x,\ C_5 = y \mid C_1 = u,\ C_2 = v,\ C_3 = w) \tag{7}$$

$$\begin{aligned} &P(C_1 = u, C_2 = v, C_3 = w, C_4 = x, C_5 = y) \\ &= P(C_1 = u) \times P(C_2 = v,\ C_3 = w \mid C_1 = u) \times P(C_4 = x,\ C_5 = y \mid C_1 = u,\ C_2 = v,\ C_3 = w) \\ &= P(C_1 = u) \times P(C_2 = v \mid C_1 = u) \times P(C_3 = w \mid C_1 = u,\ C_2 = v) \times \\ &P(C_4 = x \mid C_1 = u,\ C_2 = v,\ C_3 = w) \times P(C_5 = y \mid C_1 = u,\ C_2 = v,\ C_3 = w\ C_4 = x) \\ &= U_1[u] \times T_{3,2,1}[w,v,u] \times B_{2,1}[v,u] \times Q_{4,3,2,1}[x,w,v,u] \times P_{5,4,3,2,1}[y,x,w,v,u] \end{aligned} \tag{8}$$

ZIP Code images		GSC	ZPP
1:	5,407	87.683%	89.551%
2:	5,378	84.641%	86.575%
3:	5,533	84.312%	86.373%
4:	3,878	79.603%	81.021%

Table 3: Results on 4 different sets consistently indicate the benefits of the Markov Modeling Postprocessing (ZPP) described in this paper compared to the method that does not make use of contextual information (GSC).

Code that reflects the volume of mail received by the ZIP Code. The number of records that exist in a ZIP Code, we believe, indirectly reflect on the volume of mail "destined" for a ZIP Code. A point of future research interest would be to address the issue of sparseness in the data. Certain $n - grams$ are rare in the database and interpolation techniques must be employed to overcome this deficiency.

References

[1] T.W. Anderson, L.A. Goodman, "Statistical inference about Markov chains", in *Ann. Math Statist., 28, 89-110, 1957.*

[2] D. Bouchaffra, V. Govindaraju, and S. Srihari, "A methodology for determining probability of correctness of recognizer scores", in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, Santa Barbara, June 1998.*

[3] D. Bouchaffra, E. Koontz, V. Kripasundar and R. K. Srihari, "Incorporating Diverse Information Sources in Handwriting Recognition PostProcessing", in *International Journal of Imaging Systems and Technology, vol. 7, issue 4, Published by John Wiley, June 1996.*

[4] R.L. Dobrushin, "The description of random fields by means of conditional probabilities and conditions of its regularity", in *Theory Prob. Appl., vol. 13, pp. 197-224, 1968.*

[5] J. Favata and G. Srikantan, "A Multiple Feature/Resolution Approach to Handprinted Digit and Character Recognition", in *International Journal of Imaging Systems and Technology*, 7, pp. 304-311, 1996.

[6] V. Govindaraju, A. Shekhawat, and S. Srihari, "Determining Delivery Point Code in US mailstream", in *IWFHR III*, 1993.

[7] G. Kim and V. Govindaraju, "Bankcheck recognition using cross validation of courtesey and legal amount fields", in *International Journal of Pattern Recognition and Artificial Intelligence* 19(4), pp. 366-379, 1997.

[8] L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", in *Proceedings of the IEEE*, 77(2), pp. 257-286, 1989.

[9] Z. Shi and V. Govindaraju, "Segmentation and recognition of connected handwritten numeral strings", in *Pattern Recognition*, 30, pp. 1501-1504, 1997.

[10] R. Singhal and G. Toussaint, " Experiments in text recognition with the modified Viterbi algorithm", in *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 1(2), 1979.

[11] M. Sridhar and A. Badreldin, "Context-directed segmentation algorithm for hand-written numeral strings", in *Image and Vision Computing*, 5(1), pp. 3-9, 1997.