INFORMATICA, 1996, Vol. 7, No. 1, 3–14

DEFORMED SYSTEMS FOR CONTEXTUAL TEXT RECOGNITION

Javier ECHANOBE

Department of Electricity and Electronics
University of the Basque Country
P.O. Box 644, 48080 Bilbao, SPAIN

José R. GONZÁLEZ DE MENDÍVIL

Department of Automatic Control, Electronics and System Engineering
Public University of Navarra
Campus Arrosadia s/n, 31006 Pamplona, SPAIN

José R. GARITAGOITIA

Department of Mathematics and Computation
Public University of Navarra
Campus Arrosadia s/n, 31006 Pamplona, SPAIN

Abstract. A fuzzy method for incorporating the contextual constraints into a text recognition system is presented here. The method takes as input all the internal result that an Isolated Character Classifier (ICC) computes for an input letter, instead of an unique output character. The internal result is handled here as a fuzzy set which is then processed by a Deformed System. Such a Deformed System represents a dictionary of legal words, and it is actually a Finite Automaton which has been modified for accepting as input no single symbols but fuzzy sets. Several tests have been carried out in a Text Recognition System and the obtained results show the suitability of the method.

Key words: contextual pattern recognition, error correction, text recognition, deformed systems, fuzzy formal languages.

1. Introduction. The incorporation of contextual constraints or contextual knowledge into a text recognition system is necessary if we want to reduce the number of characters miss-recognized that an Isolated Character Classifier (ICC) produces because of several error sources in texts Fukushima *et al.* Different methods for incorporating the context can be classified as Elliman and

Lancaster (1990): (i) Statistical methods, based on the a priori knowledge of transition probabilities between characters in words and the use of the Bayes' Rule (Shinghal and Toussaint, 1979); (ii) Dictionary methods, based on the knowledge of the lexicographical constraints (in a dictionary form) of words in texts Landau (1988); and (iii) Hybrid methods, based on the use of both Statistical and Dictionary methods (Hull *et al.*, 1983; Shinghal, 1983). Hybrid methods seem to be the most efficient methods for incorporating the context into text recognition systems.

This paper deals with the problem of reducing the substitutions errors (*change-errors*) that an ICC introduces in the recognition of a text. Change errors are primarily caused by a wrong classification of the ICC. Such a problem has been treated by the above described methods and one of the best results has been obtained by using the Hybrid method proposed in Hull *et al.* (1983) (from a review in Elliman and Lancaster, 1990).

In order to achieve the reduction of change errors, another contextual post-processing method is proposed here. The method uses a dictionary for representing the lexicographical context; however, its main characteristic is the use of all the information obtained from the ICC for every input letter, instead of a unique decision (i.e., an output character). This fact allows to delay the classification of a letter until the time in which the context has been taken into account. This is the main difference with the previous methods where the classification of a letter and its correction by using the context are independent tasks. In order to implement the proposed method, the Deformed System Model (Negoita and Ralescu, 1975) is used as it has been previously introduced by the authors in Echanove *et al.* (1994) and Reina *et al.* (1992).

The method has been tested in a handprinted text recognition experiment and compared with the Hybrid method due to Hull *et al.* (1983). The results from experiments show that the recognition rates (i.e., the reduction of error by incorporating the context) obtained with the proposed method are better than those obtained with that Hybrid method. However, an important advantage is that the proposed method does not require a previous step of learning. Furthermore, the Deformed System is built automatically from the word dictionary.

The rest of the paper is organized as follows: Section 2 introduces formally the problem formulation and derives in a simple way the advantage of using a Deformed System Model. Section 3 deals with the system description and the

experimental results. Finally, Section 4 is devoted to concluding remarks.

2. Problem formulation

2.1. Isolated Character Classifier (ICC). An ICC receives an input word from a text, classifies its letters and provides an output word where one or more letters can be miss classified. Such concepts are analyzed in the following paragraphs.

Let Σ be a set of characters (i.e., an alphabet); and let Γ be a set of letters (i.e., printed characters). An ICC can be understood as a function, I, from Γ to Σ, that is, $I: \Gamma \rightarrow \Sigma$. Thus, from an input word $X = x_1 x_2 \ldots x_m$ the ICC provides an output word $Y = y_1 y_2 \ldots y_m$ where $x_i \in \Gamma$ and $y_i \in \Sigma$ being $y_i = I(x_i)$, $1 \leqslant i \leqslant m$. This process clearly preserves the length of the input word and is independent of the context in any sense; for example, functions as $y_i = I(x_{i-1}, x_i, x_{i+1})$ are not considered.

The ICC is said to introduce a *change-error* in the recognition process if there exists a character in the output word $y_i = I(x_i)$ such that x_i was not a printed y_i. In order to recuperate those *change-errors* introduced by the ICC when it recognizes a text, it is possible to use the fact that input words have some restrictions, so only certain character combinations are allowed. These restrictions are the *lexicographical context*, and can be defined by means of a dictionary of words.

Let $D \subseteq \Sigma^*$ be a dictionary where $||D|| < \infty$ and $D \subseteq \bigcup_{i:1 \ldots N} \Sigma^i$, being $N < \infty$ the length of the maximum length word in D. Therefore, the term dictionary is used for a list of words which are not associated with descriptive information such as meanings, derivations, etc. An intuitive approach for using the dictionary is achieve by means of an automaton which accepts the language defined by the dictionary. Therefore, an output word Y is said to belong to the dictionary if it is accepted by the automaton. Another approach is the search of a word in the dictionary with the minimum edit distance to the output word Y (dictionary based methods).

In order to introduce the proposed method, lets make firstly a reflection about how the ICC takes a decision when an isolated letter is presented to it. Many of the existing ICCs (Nearest Neighbour Classifiers (Cover, 1967), Bayesian Classifiers (Duda and Hart, 1973), Neural Network based Classifiers (Fukushima *et al.*, 1983), etc.), compute the *proximity* between the input isolated letter x and every one of the characters in Σ. Thus, a collection of pairs

$\{(\text{prox}(x, y_j),\ y_j \in \Sigma,\ 1 \leqslant j \leqslant ||\Sigma||\}$ is internally obtained. After that, the ICC executes a decision function and provides as output, the character with the maximum proximity value to the input letter: $\text{ICC}(x) = y_k$, being $y_k \in \Sigma \mid \forall y_j \in \Sigma: \text{prox}(x, y_k) \geqslant \text{prox}(x, y_j)$. This collection of pairs is called "internal information" of the ICC for an input character and it is used by the proposed method.

The contextual postprocessing in the dictionary based methods is clearly separated from the ICC and starts from the output word given by it. However, the proposed method uses the "internal information" provided by the ICC and processes it together with the lexicographical context. Thus, the decision function is omitted and the choice of the output character is postponed until the context is taken into account. Therefore, a special class of systems is needed in order to handle, not with single characters, but with the internal information; that is, the collection of pairs. Those systems are called Deformed Systems (Negoita and Ralescu, 1975). In the following, we introduce a Deformed System for a Finite Automaton.

2.2. Deformed System. As it was mentioned above, the natural way for implementing a dictionary is by means of a State Automaton. Because the dictionary is finite, the automaton is finite and deterministic (Hopcroft *et al.*, 1979). Furthermore, the implementation of the finite state automaton from the dictionary is an automatic process.

Let $A \equiv (\Sigma, Q, q_0, \delta, \lambda, \Delta)$ be a Moore finite deterministic automaton which accepts the dictionary D; the elements of the automaton are defined as: (i) Σ is the alphabet, $\Sigma = \{y_1, y_2, , \ldots, y_r, \ldots\}$ where y_i $(i = 1, 2, \ldots, r \ldots)$ are characters; (ii) Q is the set of states; (iii) $q_0 \in Q$ is the initial state; (iv) δ is the transition function defined by a graph $\delta: Q \times \Sigma \times Q \rightarrow \{0, 1\}$, being $\delta(q, y, q') = 1$ if the transition from q to q' by the character y exists, and $\delta(q, y, q') = 0$ if such transition does not exists; (iv) Δ is the output alphabet, where $\Delta \cup \{\varepsilon\}$ (ε denotes the empty string); (v) λ is the output function defined as $\lambda: Q \rightarrow \Delta$.

Given a particular dictionary D, the elements above presented can be calculated as follows. Let Q be a finite set of states; initially $\forall q, q' \in Q$ and $y \in \Sigma:\ \delta(q, y, q') = 0$ and $\lambda(q) = \varepsilon$. Given a sequence $w = y_1 y_2 \ldots y_t \ldots y_m$ of characters in Σ such that $w \in D$, a sequence of states $q_0 \ldots q_t \ldots q_m$ is defined via construction, where $\delta(\cdot, \cdot, \cdot)$ is updated by $\delta(q_{t-1}, y_t, q_t) = 1$ with

$1 \leqslant t \leqslant m$. Finally, $\lambda(q_m) = w$. The method is constrained for obtaining a deterministic automaton. This automaton is able now, to determine, if a given word $Z = z_1 z_2 \ldots z_t \ldots z_n$ belongs or not to the dictionary. For doing that, the automaton performs transitions with the characters $z_1 z_2 \ldots z_n$ an reaches an state, say q_n. Then, if $\lambda(q_n) = Z$, then Z belongs to the dictionary; otherwise $(\lambda(q) = \varepsilon)$ Z is rejected.

A fuzzy interpretation. As it has previously shown, for an input letter x, the ICC provides the internal information as a collection of pairs $\{(\text{prox}(x, y_j), y_j)\}$. By making $\text{prox}(x, y_j) \in [0, 1]$, $\forall y_j \in \Sigma$, and due to the fact that these values represent the similarity of x to the characters y_j, then the collection of pairs can be interpreted as a fuzzy set $\tilde{x}$ in the universe Σ. Values $proximity(x, y_j) = \mu_{\tilde{x}}(y_j)$ define the membership function of the fuzzy set $\tilde{x}$. Therefore the ICC provides for an input letter x, a fuzzy set $\tilde{x} = \{(y_j, \mu_{\tilde{x}}(y_j))\}$, called fuzzy character (see Fig. 1).

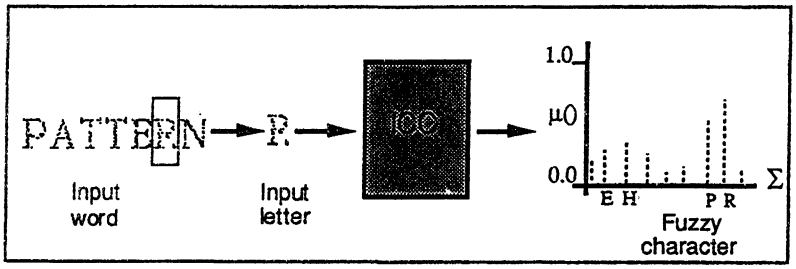

Fig. 1. The ICC provides a fuzzy character for an input letter.

In order to handle with fuzzy characters instead of single symbols, the automaton A must be modified for accepts as inputs, fuzzy sets. By doing this, a Deformed System is obtained (Negoita and Ralescu, 1975). The Deformed System for the automaton A is defined as the tuple $A^D \equiv ((\Sigma, \tilde{x}), (Q, S), q_0, \delta, \lambda, (\Delta, T))$ where Σ, Q and Δ are fuzzily constrained by the fuzzy sets $\tilde{x}, S$ and T respectively; δ is the transition function whose domain is fuzzily constrained by $S \times \tilde{x} \times S$; λ is the output function whose domain is fuzzily constrained by S and its range is restricted by T. Consider the generic fuzzy sets $\tilde{x}, S$ and T defined as $\tilde{x} = \{(\mu_{\tilde{x}}(y), y),\ y \in \Sigma\}$; $S = \{(\mu_S(q), q),\ q \in Q\}$; $T = \{(\mu_T(w), w),\ w \in \Delta\}$ with universes Σ, Q and Δ respectively. The state

equations of the Deformed System are the following (t denotes the t-th step):

$$\text{Given} \quad \delta(q_t, y_t, q_{t+1})$$
$$\text{then} \quad \mu_S(q_{t+1} \geqslant \min(\mu_S(q_t), \mu_{\tilde{x}}(y_t), \delta(q_t, y_t, q_{t+1})). \tag{1}$$

$$\text{Given} \quad w = \lambda(q_t) \quad \text{then} \quad \mu_T(w) \geqslant \mu_S(q_t). \tag{2}$$

The behaviour of the ICC together with the Deformed System is as follows (see Fig. 2). Given an input word $X = x_1 x_2 \ldots x_m$, the ICC provides a fuzzy character string $\tilde{x}_1 \ldots \tilde{x}_m$. After this, the Deformed System takes every one of the fuzzy characters and makes transitions with them. Because of the input elements are fuzzy sets and no single symbols, there are multiple transitions in the Deformed System, and therefore a fuzzy set of states is reached (i.e., every state with its membership value). Now, the output function λ is applied for every state and a fuzzy set $T = \{(\mu_T(w), w),\ w \in \Delta\}$ is obtained. The final output word for X is the word $Y \in D$ such that it verifies $\mu_T(Y)$ is the largest membership value in T.

The computation of the fuzzy set T which is the output of the Deformed System, is achieved simultaneously from the equations (1) and (2). However, the state equation (1) can be very restrictive due to the fact that characters y_t whose values of $\mu_{\tilde{x}}(y_t)$ are small, dominate in the computation of the set T (because of the min operator). In order to reinforce the characters with high values in their membership functions, during the computation of T, it is necessary to use another state equations which reflect this aspect (i.e., an average composition). The modification of the Deformed System equations is possible due to the flexibility of the definition for such systems. Thus, these other equations are proposed:

$$\text{Given} \quad \delta(q_t, y_t, q_{t+1})$$
$$\text{then} \quad \mu_S(q_{t+1} \geqslant \frac{\mu_S(q_t) \cdot t + \mu_{\tilde{x}}(y_t)}{t+1} \cdot \delta(q_t, y_t, q_{t+1}). \tag{3}$$

$$\text{Given} \quad w = \lambda(q_t) \quad \text{then} \quad \mu_T(w) \geqslant \mu_S(q_t). \tag{4}$$

We claim that for a set of input words, the set of output words obtained by the proposed method has less words with change-errors than the set of output words provided by the ICC. In the following section, an experiment is developed in order to prove this fact. Furthermore, the method is compared with another method for contextual postprocessing.

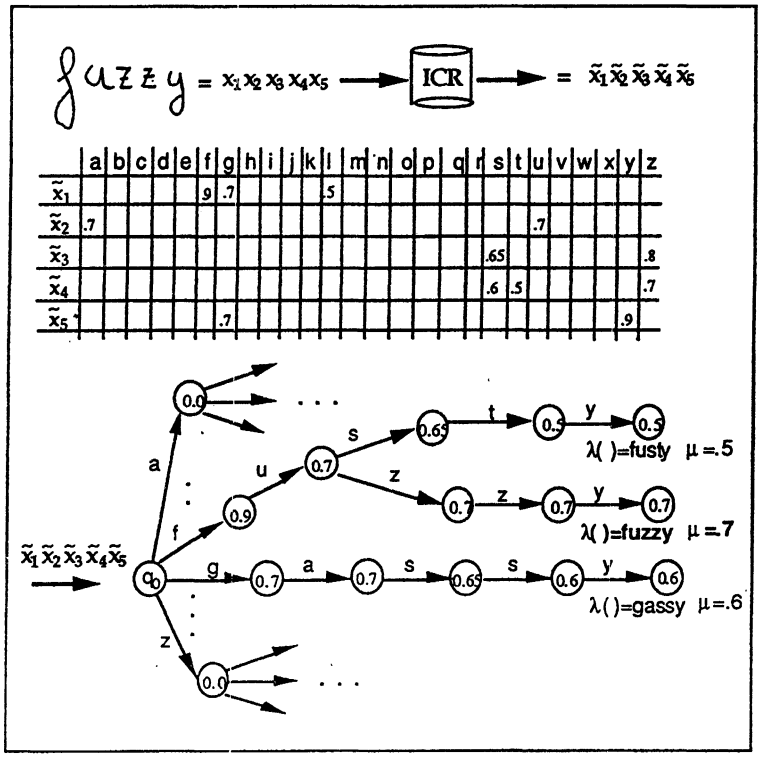

Fig. 2. Recognition of a word by the ICC and the Deformed System.

3. Experimental results. In order to test the performances of the proposed method, an experimental process intended to reduce the number of words with *change-errors* introduced by an ICC in the recognition of a text, has been carried out. This goal is achieved by incorporating the context with a Deformed System in the way showed in the previous sections.

In the experiment, the ICC is a Neural Network: a Perceptron with two layers. The output layer has 26 units and each one is in correspondence with a character in the alphabet. The Neural Network was previously trained with 26 prototype letters. When an input letter is processed by the Neural Network, the value of an output unit (in the real interval $[0, 1]$) represents the proximity between the input letter and the character associated with that unit. According

to this fact, the Neural Network provides as output character, the character associated with the largest value output unit. However, instead of taking such an output, a fuzzy character with all the output unit values is built as it has been explained; that is, for an input letter x, the Neural Network produces a fuzzy set $\tilde{x} = \{(\mu_{\tilde{x}}(y), y),\ y \in \Sigma\}$ where Σ is the alphabet and $\mu_{\tilde{x}}(y)$ is the value of the output unit associated with the character y. Note that a change-error is produced by the ICC when, for an input letter x which is a printed representation of the character y, the value $\mu_{\tilde{x}}(y)$ is not the largest membership value in the fuzzy character $\tilde{x}$.

In order to control the number of *change-errors* introduced by the ICC and in consequence to make an extensive evaluation of the method capabilities, a simulation of the ICC recognition process for a given text, has been performed. Such a simulation process is described in the following paragraphs.

First of all, a database of isolated upper-case handwritten letters has been created. This database is composed of 100 handwritten letters for each character in the alphabet (i.e., 100 "A" letters, 100 "B" letters, etc.); therefore, 2600 letters are obtained. Each one of these letters is processed by the ICC for obtaining a collection of fuzzy sets, called fuzzy characters (100 fuzzy characters for the "A", 100 for the "B", etc.). In a group of fuzzy characters, for instance, the group of 100 fuzzy characters for the 'A', some of them would eventually produce a *change-error* because of the ICC miss-recognizes a percentage of the letters. These fuzzy characters are called erroneous fuzzy characters .

Once the collection of fuzzy characters (correct and erroneous fuzzy characters) has been obtained, each letter in the text to be recognized is replaced by one of its respective fuzzy characters. This process is achieved in a randomize way, so anyone of the respective fuzzy characters can be the substitute. However, the process is constrained in order to introduce a desired percentage of words with erroneous fuzzy characters. In what follows, we refer to this percentage of error introduced by the ICC as the Change-Error Rate:

$$\text{Change-Error Rate} = \frac{\text{number of words with erroneous fuzzy characters}}{\text{number of words in the text}} \times 100.$$

Thus, the behaviour of the ICC when it recognizes an input text and provides for an input letter a fuzzy character instead of an output character, is simulated.

For this experiment, we have selected a computational domain text that contains 6396 words, and a dictionary with 1700 words. The reason for selecting these dictionary and text sizes is to compare the method with the Hybrid method proposed in Hull *et al.*, (1983), where similar sizes were used. Furthermore, 1700 words can summarize 75 percent of written English (Kucera and Francis, 1967). The results obtained are shown in Table 1.

Table 1. Experimental results for different Change-Error Rates

Change-Error Rate	83.9	75.17	68.27	46.01	30.94	23.24
Recognition Rate	96.71	97.58	97.92	98.75	99.10	99.12
Error-Reduction Rate	96.08	96.78	96.95	97.28	97.09	97.50

The Change-Error Rate is produced by the Neural-Network.

The Error-Reduction Rate (reduction of *change-error* words) is achieved by the Deformed System):

$$\text{Error-Reduction Rate} =$$
$$= 100 - \left[\frac{\text{number of miss recognized words in the output text}}{\text{Change-Error Rate}} \times 100\right].$$

The Recognition Rate is obtained with the complete system (ICC+Deformed System).

It can be observed how the Error-Reduction Rates are very high for all the Change-Error Rates. Thus for example in the case of 30.94% of Change-Error Rate, the method reduces the error in the 97.09%. This value is higher than the 87% obtained in the Hull-Srihari-Choudhari experiment (Hull *et al.*, 1983) for a 31% of Change-Error Rate.

4. Conclusions. A fuzzy method for incorporating the context into a Text Recognition System has been proposed. This method is based on a Deformed System which is built automatically from a dictionary of legal words. The method is easy to implement and does not require previous training. The obtained results in a handprinted text recognition experiment, show good performances of the method. A comparison with one of the best Hybrid methods

has been presented, and the results show an improvement with respect to it for a problem of *change-error* reduction.

Due to the flexibility of the Deformed System, it is possible to develop another composition functions for particular problems. Furthermore, it is possible in a simple way to join the method with syntactic context based on grammars. A future perspective is the use of the proposed method to deal with the problem of insert and delete errors in printed texts.

Acknowledgements. The authors wish to thank the Basque Government for providing the financial support of this work.

REFERENCES

Cover, T.M., and P.E. Hart (1973). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory,* 1321–1327.

Duda, R.O., and P.E. Hart (1973). *Pattern Classification and Scene Analysis.* Addison-Wesley, New York.

Echanove, J., R. Reina, J.R. Garitagoitia and J.R. González de Mendívil (1994). Deformed systems in text recognition. *International Conference On Artificial Neural Networks,* Sorrento (Italy), May 26–29.

Elliman, D.G., and I.T. Lancaster (1990). A review of segmentation and contextual analysis techniques for text recognition. *Pattern Recognition,* **23**, 337–346.

Hopcroft, J., and J. Ullman (1979). *Introduction to Automata Theory, Languages and Computation.* Addison-Wesley Publishing Company, Reading Massachusetts.

Hull, J.J., S.N. Srihari and R. Choudhari (1983). An integrated algorithm for text recognition: comparison with a cascaded algorithm. *IEEE Trans. Pattern Analysis Mach. Intell.,* **5**, 584–395.

Fukushima, K., S. Miyake and T. Ito (1983). Neocognitron: A Neural network for a mechanism of visual pattern recognition. *IEEE Trans. Sys. Man and Cyber.,* **13**, 826–834.

Landau, G.M. (1988). Fast string matching with k differences. *J. Comput. Syst. Sci.,* **37**, 63–78.

Negoita, C.V., and D.A. Ralescu (1975). *Application of Fuzzy Sets to System Analysis.* Birkaeuser, Basilea.

Reina, R. José R. González de Mendívil, José R. Garitagoitia (1992). Improved character recognition System based on a Neural Network incorporating the context via Fuzzy Automata. *2nd International Conference on Fuzzy Logic & Neural Networks,* Izuka-Japan, July 17–22.

Shinghal, R., and G.T. Toussaint (1979). Experiments in text recognition with the modified viterbi algorithm. *IEEE Trans. Pattern Analysis Mach. Intell.*, **1**, 184–193.

Shinghal, R. (1983). A hybrid algorithm for contextual text recognition. *Pattern Recognition*, **16**, 184–193.

Kucera, H., and W.N. Francis (1967). *Computational Analysis of Present-Day American English.* Providence, RI: Brown Univ. Press.

Received October 1995

J. Echanobe was born in 1965. He graduated from the University of the Country Basque in 1990 in physics, where he is currently obtaining his degree of Doctor in Physics, at the Department of Electricity and Electronics. His main research interests include pattern recognition, formal languages theory, fuzzy sets theory, text recognition.

J.R. González de Mendívil was born in 1963. He received the degree of Physicist (area: Electronics and Control) from the University of the Basque Country in 1988, and the degree of Doctor in Physics from the same university in 1993. He is an Associated Professor in the Department of Automatic Control, Electronics and System Engineering in the Public University of Navarra. His research interests are the problems of pattern recognition, the dead-lock detection in computing systems and the simulation and modelling of dynamical systems.

J.R. Garitagoitia was born in 1953. He received the degree of Physicist (area: Electronics and Control) from the University of the Basque Country in 1976, and the degree of Doctor in Physics from the same university in 1981. He is a Professor in the Department of Mathemathics and Computation in the Public University of Navarra. His research interest include formal languages theory, fuzzy sets theory, text recognition, dead-lock detection in computing systems and the simulation and modelling of dynamical systems.

KONTEKSTINIO TEKSTŲ ATPAŽINIMO DEFORMUOTOS SISTEMOS

Javier ECHANOBE, José R. GONZALEZ DE MENDIVIL,
José R. GARITAGOITIA

Straipsnyje glaustai aptariamas konteksto panaudojimas (įjungimas) tekstų atpažinimo sistemose. Daugiausia dėmesio skiriama keitimo klaidų (*change-errors*), sumažinimo problemai išspręsti (sąlygas joms atsirasti sudaro izoliuoto simbolių klasifikatoriaus (ICC) naudojimas tekstų atpažinimo sistemose). Aprašomo metodo rezultatai palyginami su Hibridinio metodo (dažniausiai taikomo tekstams atpažinti) rezultatais.

Straipsnis gali sudominti specialistus, nagrinėjančius ranka rašyto teksto atpažinimo problemas, juolab, kad eksperimentas, kurio rezultatais remiamasi, parodo, kad aprašytą metodą lengva įdiegti ir jis nereikalauja išankstinio pasiruošimo.