ISSN: 2229 – 6956(ONLINE)
DOI: 10.21917/ijsc.2011.0025

FINGERPRINT CLASSIFICATION BASED ON RECURSIVE NEURAL NETWORK WITH SUPPORT VECTOR MACHINE

T. Chakravarthy[1], K. Meena[2] and D. Nathiya[3]
A. V. V.M Sri Pushpam College, Bharathidasan University, Tamil Nadu, India
E-mail: [1]tcvarthy@gmail.com, [3]nathiyad24@yahoo.co.in
[2]Shrimati Indira Gandhi College, Bharathidasan University, Tamil Nadu, India

Abstract
Fingerprint classification based on statistical and structural (RNN and SVM) approach. RNNs are trained on a structured representation of the fingerprint image. They are also used to extract a set of distributed features of the fingerprint which can be integrated in this support vector machine. SVMs are combined with a new error correcting codes scheme. This approach has two main advantages. (a) It can tolerate the presence of ambiguous fingerprint images in the training set and (b) It can effectively identify the most difficult fingerprint images in the test set. In this experiment on the fingerprint database NIST-4 (National Institute of Science and Technology), our best classification accuracy of 94.7% is obtained by training SVM on both fingerCode and RNN –extracted futures of segmentation algorithm which has used very sophisticated "region growing process".

Keywords:
Support Vector Machine, Recursive Neural Network, Region Growing, Error Correction Code

1. INTRODUCTION

Today, the fingerprint is the pillar of modern criminal identification. Fingerprint patterns variations are within a limit which allows a systematic classification of these configurations. The real significance of a fingerprint patterns is based mainly on the following principles.

Unchangebility: The configuration and details of the patterns or permanent and near change throughout ones life until the skin disintegrates after death.

Uniqueness: The degree of variation of the ridges is so high that two patterns with the same characteristics never occur either in another finger of the same person (or) in a different individual. Hentry's Classification consists of three major types. Archs, loops and Whorls [1],[2].

Arch

1. Plain Arch
2. Tented Arch

Loop

1. Left Loop
2. Right Loop

Whorl

1. Plain Whorl
2. Central Pocket Whorl
3. Double loop Whorl
4. Accidental Whorl

Plain Arch

Fingerprint patterns in which he ridges enter on one side, rise in the middle, and flow (or) tend to flow out from the other side.

Tented Arch

The same tendency to enter from one side and flow out from the other side, with the exception that the ridges from either and angle.

Left Loop

Loops whose ridges flow in the direction of radial bone (toward the thumb finger) are called left loop.

Right loop

Loops whose ridges flow in the directions of ulnar bone (toward the little finger) are called Right loop.

Plain Whorl

Any patterns with at least two deltas and one re curving ridge which may be a spiral (or) any variation of a circle is called a plain whorl.

Central Pocket Whorl

In the central pocket loop which has two deltas and at least a ridge making a complete circuit as in the plain loop, the imaginary line drawn between the two deltas must not touch any of the recurve ridges within the pattern area.

Double Loop Whorl

The double (or) twinned loop; consist of two deltas and two separate loops. the separate and distinct shoulders do not imply that the ridges are disconnected.

Accidental whorl

The accidental whorl is a pattern consisting of a combination of two (or) more different types of patterns, with the exception of the plain arch, with two (or) more deltas.

Contributions:

Recursive Neural Networks and support Vector Machines:

The fingerprint classification by the segmentation algorithm is based on two machine learning:

i. Support Vector machines(SVMs) and
ii. Recursive Neural Networks(RNNs)

Recursive Neural Networks are trained on a structured representation of the fingerprint image. The pattern recognition for a two class problem is made by determining the separating hyperplane that has maximum distance to the closest points of the training set. These closet points are called support vector. If the two classes are non-separable it can be still looked for the hyperpalne that maximizes the margin and minimizes a quantity proportional to the number of misclassification errors.

Support Vector Machine is a relatively new technique for pattern classification and regression that is well founded in statistical learning theory [3]. One of the main attractions of using Support Vector Machines is that they are capable of learning in sparse high-dimensional spaces with very few training examples. They have been successfully applied to various classification problems [4]. An Recursive Neural Network is a connectionist architecture designed for solving the supervised learning problem when the instances space is comprised of labeled graphs [5]. This architecture can explain structural information in the data, which as explained above, may help in discriminating between certain classes. In this paper, Recursive Neural Networks are also used to extract a distributed vectorial representation of the relational graph associated with a fingerprint. This vector is regarded as an additional set of features subsequently used as inputs for the Support Vector Machines classifier.

An important issue in fingerprint classification is the problem of ambiguous examples; some fingerprints are assigned to two classes simultaneously, i.e they have double labels (these images are also called "cross-referenced"). In order to address this issue, an error-correcting code [6] scheme of Support Vector Machine classifiers based on a new type of decoding distance has been used. First it allows a more accurate use of ambiguous examples because each Support Vector Machine is in charge of generating only one code bit, whose value discriminates between two disjoint sets of classes. Then, if a fingerprint has labels all blanking to the same set dor a particular code bit, we can retain this example in the training set without introducing any labeling noise. The second advantage of this system is the capability to deal with rejection problems. This is due to the concept of margin inherent to the Support Vector Machine, which is incorporated in the decoding distance.

Related Work:

Several approaches have been developed for automatic fingerprint classification. These approaches can be broadly categorized into five main kinds.

1. Model based
2. Structure based
3. Frequency based
4. Syntactic based
5. Hybrid Approaches

1. Model Based

The model based fingerprint classification technique uses the locations of singular points (core and delta) to classify a fingerprint into one of the five classes [7], [8], [9]. It tries to capture the knowledge of a human expert by deriving rules for each category by hand, constructing the models and therefore, does not require training.

2. Structure Based

A structure based approach uses the estimated orientation field in a fingerprint image to classify the fingerprint into one of the five classes. The neural network used in [10] was trained on images from NIST-4 databases. A similar structure based approach, which uses hidden markov models for classification[11], depends on a reliable estimation of ridge location, which is difficult in noisy images.

In another structure based approach B-Spline curves are used to represent and classify fingerprints [12].

3. Syntactic based

Syntactic approach uses formal grammar to represent and classify fingerprints [13].

4. Frequency based

Frequency based approach uses the frequency spectrum of the fingerprints for classification [14],[15].

5. Hybrid Approaches

Combine two or more approaches for fingerprint classification [16],[17].

The remaining part of this paper is organized as follows in section 2 illustrates our proposed work. In section 3 contains in our algorithm, and section 4 presents implementation and results. In Section 5 comparative analyses. Finally, we report our conclusion in section 6.

2. PROPOSED APPROACH

This work presents three series of experiments, as a base fingerprint representation is used for fingerCode features and a statistical representation scheme is proposed [18]. In the first set of experiments fingerCode features are combined with structural representations of a fingerprint based on relational graphs. In this case connectionist architecture integrates statistical and structural representations.. In the second experiments Support Vector Machines are trained on fingerCode [19] preprocessed images. This result is more accurate than the result obtained in [20] using the same features and a two stage K-Nearest Neighbor / MultiLayerPerceptron classifier. Interestingly Support Vector Machine accuracy is much better than separate accuracies of both K- Nearest Neighbor and MultiLayerPerceptron. Finally the Support Vector Machine is trained on both finger code and Recursive Neural Network extracted features. In doing so, the performance is improved and the accuracy rate also increases.

Classification schemes based on training one-vs-all and pairwise classifier have two extreme approaches; the first uses all the data, the second the smallest portion of the data. In practice, it can be more effective to use intermediate classification strategies in the style of error correcting codes.

In this case each classifier is trained to separate a subset of classes from another disjoint subset of classes. For example the first set could consist of classes A and T and the second of classes R,L and W(PW,DW,CW) by doing so, each of the class is associated with a row of the "coding matrix" M€{-1,0,1) qxs, where s denotes the number of classifiers. M_{ij}=-1 or 1 indicates that points in class i are regarded as negative (or) Positive examples for training the j^{th} classifier. M_{ij}=0 indicates that points in class i are not used for training the j^{th} classifier.

3. ALGORITHM

1. Estimate the orientation field O using the least square orientation estimation algorithm. Orientation field O is defined as an NxN image, where O(i,j). an image is divided into a set of ωxω non overlapping windows and a single local orientation is defined for each window.
2. Smooth the orientation field in a local neighborhood. Let the smoothed orientation field be represented as O`.
3. Initialize A, a label image used to indicate the core point.
4. For each pixel (i,j) in O` compute the poincare index and assign the corresponding pixel in A a value of one if the poincare index is (1/2). The poincare index a pixel (i,j) enclosed by a digital curve, which consist of a sequence of pixels that are on (or) within a distance of one pixel apart from the corresponding curve is computed as follows.

$$\text{Poincare}(i,j) = \frac{1}{2\pi}\sum_{k=0}^{N\Psi-1}\Delta(K),$$

$$\Delta(K) = \begin{cases} \delta(k), & \text{if } |\delta(k)| < \frac{\pi}{2} \\ \pi + \delta(k), & \text{if } |\delta(k)| \le -\frac{\pi}{2} \\ \pi - \delta(k), & \text{otherwise} \end{cases}$$

$$\delta(k) = O'\left(\Psi_x\left(k'\right), \Psi_y\left(k'\right)\right) - O'\left(\Psi_x(k), \Psi_y(k)\right)$$

$$k` = (k+1) \bmod N\psi,$$

Where $\Psi_x(.)$ and $\Psi_y(.)$ are the x and y coordinates of the closed digital curve with Nψ pixels.

5. Find the connected components in A. If the area of a connected component is larger than seven, a core is detected at the centroid of the connected component. If the area of the connected component is larger than 20, two cores are detected at the centroid of the connected component.
6. If more than two cores are detected go back to step 2
7. If two cores are detected the center is assigned the coordinates of the core points with the lower y value (the upper core). If only one core is detected. The center is assigned the coordinates of the core point.
8. If no core point is detected, compute the covariance matrix of the vector field in a local neighborhood (qxq) of each point in the orientation field shown in Fig.1.

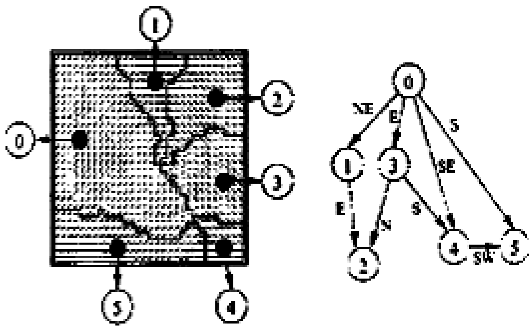

Fig.1 a) Segmented fingerprint image b) Relational Graph

4. IMPLEMENTATION AND RESULTS - RNN and SVM Method

4.1 VECTOR-BASED AND STRUCTURAL CLASSIFICATION

The used models have trained a multiplayer perceptron using the fingercode feature vector as input. Fingercode feature consists of a vector of 192 real features computed in three steps. First, the fingerprint core and centre are located. Then the algorithm separates the number of ridges present in four directions (0^0, 45^0, 90^0 and 135^0) by filtering the central part of the fingerprint with a bank of Gabor filters. Finally standard deviations of grayscale values are computed on 48 disk sectors for each of the four directions is shown in Fig.2.

Fig.2. Segmentation of 48 sectors

The best performance on the test set obtained with Multilayerperceptron architecture with 20-40 hidden units, 192 input units and 7 output units is shown in Fig.3.

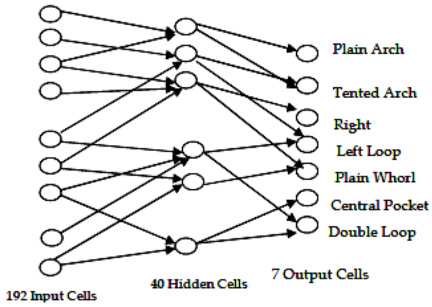

Fig.3. Neural Network Architecture

Table.1 shows the corresponding confusion matrix. The NIST-4 database consists of 4000 fingerprint images from 2000 fingers with two impressions for each finger. The first impression is used for the test image set which contains 400 fingerprint images for each of the five classes. Since the model is designed for seven classes problem, the database is divided into 312, 25 and 63 images for PW, CW, and DW classes respectively. The overall accuracy is 93.3% shown in Table.1.

Table.1. Multilayerperceptron Using FingerCode

True Class		Assigned Class							
Type	Number	PA	TA	RL	LL	PW	CW	DW	%
PA	400	380	9	6	5	0	0	0	95.0
TA	400	6	386	4	4	0	0	0	96.5
RL	400	2	4	382	12	0	0	0	95.5
LL	400	2	2	80	378	0	0	0	94.5
PW	312	0	0	0	0	294	14	4	94.2

CW	25	0	0	0	0	2	23	0	92.0
DW	63	0	0	0	0	3	6	54	85.7

(653.4/7=93.3%)

The confusion matrix of the Recursive Neural Network using relational graphs trained on the structural features. In this case the overall accuracy is 94.25% (659.8/7=94.25) shown in Table.2.

Table.2. Recursive Neural Network Using Relational Graph

True Class		Assigned Class							
Type	**Number**	**PA**	**TA**	**RL**	**LL**	**PW**	**CW**	**DW**	**%**
PA	400	384	12	4	0	0	0	0	96.0
TA	400	9	387	2	2	0	0	0	96.7
RL	400	2	4	382	12	0	0	0	95.5
LL	400	0	4	12	384	0	0	0	96.0
PW	312	0	0	0	0	296	4	2	94.0
CW	25	0	0	0	0	2	23	0	92.0
DW	63	0	0	0	0	4	3	56	88.8

(659.8/7=94.25%)

Point out that accuracy of the structural classifier is much lower than the one of the statistical classifier. This is mainly due to the large degree of confusion matrix among plain whorl (PW), Central pocket whorl (CW), Left loop(LL) and Right Loop(RL) classes. On the other hand, as expected the best performance of the structural classifier is a related to the discrimination between plain arch and whorl classes. A k-nearest neighbor classifier has been used for combining the statistical and Structural classifiers. Table.3. dipicts the confusion matrix of Multilayerpercetron and Recursive Neural Network for this experiment. The accuracy of this classifier is 95.2 %(666.4/7=95.25%).

4.2 RESULT OF SUPPORT VECTOR MACHINE

The Support Vector Machine method is used with three types of multiclass classification. i.e. one-vs-all Support Vector Machine, pair wise Support Vector Machine, Error Correcting Code.

The accuracy in results of various, approaches of Support Vector Machine is given as below;

i	One vs all SVM	95.3%	Table.4
ii	Pair wise SVM	94.08%	Table.5
iii	ECC SVM with Margin Weighted Euclidean Decoding	95.23%	Table.6
iv	Hamming distance code	94.2%	
v	Soft margin distance	95.52%	

Table.3. Multilayerperceptron and Recursive Neural Network

True Class		Assigned Class							
Type	**Number**	**PA**	**TA**	**RL**	**LL**	**PW**	**CW**	**DW**	**%**
PA	400	386	8	4	2	0	0	0	96.5
TA	400	6	389	2	3	0	0	0	97.0
RL	400	1	3	388	8	0	0	0	97.0
LL	400	0	3	8	389	0	0	0	97.1
PW	312	0	0	0	0	301	7	4	96.4
CW	25	0	0	0	0	2	23	0	92.0
DW	63	0	0	0	0	2	4	57	90.4

(666.4/7=95.2%)

Table.4. One-vs-all Support Vector Machine

True Class		Assigned Class							
Type	**Number**	**PA**	**TA**	**RL**	**LL**	**PW**	**CW**	**DW**	**%**
PA	400	392	4	2	2	0	0	0	98
TA	400	8	388	3	1	0	0	0	97
RL	400	1	2	391	6	0	0	0	97.4
LL	400	2	0	4	394	0	0	0	98.5
PW	312	0	0	2	0	302	6	20	96.7
CW	25	0	0	0	0	2	22	1	88.0
DW	63	0	0	0	0	1	4	4	92.0

(667.6/7=95.3%)

Table.5. Pairwise Support Vector Machine

True Class		Assigned Class							
Type	**Number**	**PA**	**TA**	**RL**	**LL**	**PW**	**CW**	**DW**	**%**
PA	400	390	6	2	1	0	1	0	97.5
TA	400	9	387	3	1	0	0	0	96.7
RL	400	0	2	386	10	0	2	0	96.5
LL	400	0	1	6	392	1	0	0	68.0
PW	312	0	0	2	3	298	6	3	95.5
CW	25	0	0	1	0	1	21	2	84.0
DW	63	0	1	0	1	2	2	57	90.4

(658/7=94.08%)

Table.6. Error Correcting Code Support Vector Machine with Margin-Weighted Euclidean Decoding

True Class		Assigned Class							
Type	**Number**	**PA**	**TA**	**RL**	**LL**	**PW**	**CW**	**DW**	**%**
PA	400	388	6	3	2	0	1	0	97.0
TA	400	4	391	2	2	1	0	0	97.7
RL	400	0	2	385	12	0	1	0	96.01
LL	400	1	1	3	394	1	0	0	97.1
PW	312	1	0	2	2	296	8	3	94.8
CW	25	0	0	0	0	1	23	1	92.0
DW	63	0	0	0	1	2	2	58	92.0

(666.6/7=95.25%)

In the one – vs –all Support Vector Machine method the accuracy of classification is higher than in the pairwise method. Pairwise method and Hamming distance code method gives more (or) less the same results. Similarly the results are nearly the same in error correction code support vector machine method and soft margin distance method. These observations are shown in Table.4, 5 and 6. The work has trained Support Vector Machine on both fingercode and Recursive Neural Network extracted features and used the Error Correction code scheme with margin weighted Euclidean decoding. The confusion matrix is summarized in Table.7. The performance is improved to 96.0% (672.0/7=96.0%). The performance is improved from 93.0% to 96.0% when the results of fingercode method (MultiLayerPerceptron) and Support Vector Machine with Recursive Neural Network method are compared the Support Vector Machine with Recursive Neural Network method gives higher results.

Table.7. Error Correcting Code of Support Vector Machine trained on both Finger Code and Recursive Neural Network

True Class		Assigned Class							
Type	**Number**	**PA**	**TA**	**RL**	**LL**	**PW**	**CW**	**DW**	**%**
PA	400	392	4	2	1	0	1	0	98.0
TA	400	6	390	0	2	1	1	0	97.5
RL	400	0	2	393	4	1	0	0	98.1
LL	400	0	1	2	396	0	1	0	99.0
PW	312	0	2	1	2	300	4	2	95.8
CW	25	0	0	0	1	1	23	0	92.0
DW	63	0	1	1	0	0	2	59	93.6

(672.0/7=96.0%)

5. COMPARATIVE ANALYSIS

Sl. No.	Techniques		No of Classification	Accuracy
Previous Method				
1	Model Based		Five classes (Arch, Tented arch, left loop, Right loop and whorl)	85%
2	Structure Based	1.Artificial Neural Network	Five classes	90.2%
		2.B-Spline	Five classes	88.9%
3	Rich Distribution with NFA		Five Classes	96%
4	Frequency Based		Five Classes	94.8%
5	Syntactic based formal grammer		Five Classes	90%
6	Hybrid Approach		Five Classes	87.5%
Proposed Method				
1	Multilayer Perceptron using finger Code		Seven Classes (Arch, Tented Arch, Left loop, Right loop, Plain Whorl, Central Pocket Whorl, Double loop Whorl)	93.3%
2	Recursive Neural Network using Relational Graph		Seven Classes	94.25%
3	Multilayer perceptron and recursive Neural Network		Seven Classes	95.2%
4	One – vs-all support vector machine		Seven classes	95.3%
5	Pairwise support vector machine		Seven classes	94.08%
6	Error correction code Support Vector Machine with margin weighted Euclian decoding		Seven classes	95.25%
7	Error Correction code of Support Vector Machine trained on both fingercode and recursive neural network		Seven classes	96.0%

6. CONCLUSION

In this paper we have studied the combination of statistical and structure representation for fingerprint classification. On algorithm for extracting a structural representation of fingerprint images is presented recursive neural network are used to process this structural representation and to extract a distributed vectorial representation of the fingerprint. This study differs from previous related work by taking into consideration of seven different classes instead of five different classes. The advantage of this work lies not only on the decision making. but also, its ability to perform multiple classifications upto seven classes. The support Vector Machine with Recursive Neural Network method shows better accuracy than fingercode method (Multilayer Perceptron).

REFERENCES

[1] C.E. Chapel, 1941, “Fingerprinting – A manual of identification”, Coward-Mccann.

[2] FBI’s Manual, 1963, “The science of fingerprints U.S. Government printing office”.

[3] V.N Vapnik 1998 “Statistical learning theory” Wiley, New York.

[4] N. Cristianini, J. Shawe-Taylor, 2006 “An Introduction to Support Vector Machines”, Cambridge University Press, Cambridge.

[5] P. Frasconi, M. Gori. A. Sperduti, 2008 “A general framework for adaptive processing of data structures”, IEEE trans. Neural Networks, Vol.9, No.5, pp.768-786.

[6] T.G Dietterich, G. Bakiri, 2009, “Solving Multiclass learning problems via error-correcting output codes”, J. Artificial Intelligence.

[7] A.K Hrechak and J.A McHngh, 2008 “Automated fingerprint recognition using structural Matching”, Pattern Recognition, Vol.23, No.8, pp.893- 904.

[8] K .Karu and A.K.Jain, 1996, “Fingerprint Classification Pattern Recognition”. Vol.29, No. 3, pp. 389-404.

[9] L.Hong and A.K.Jain, 1998, “Clasification of Fingerprint Images”, Technical Report MSUCPS: TR 98-18, Michigan state University.

[10] C.I Waston, 1993, “NIST Special Database 9: Mated Finger card pairs”, Advanced Systems Division, Image Recognition Group, National Institute of Science and Technology.

[11] A.Senior, 1997, “Hidden Markov Model Fingerprint Classifier”, proc. 31st Asilomar Conf. Signal Systems and Computers, pp.306-310.

[12] D.K Isenor and S.G Zaky, 2009 “Fingerprint identification using graph matching”, Pattern recognition, Vol.19, No.2, pp.113-122.

[13] C.V.K Rao and K. Black, 2009 “Type Classification of fingerprints: A Syntactic approach”, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-2, No. 3, pp.223-231

[14] B. Moayer and K.S Fu, 2007, “An Application of Stochastic Languages to fingerprint pattern recognition”, Pattern recognition, Vol.8, No.3, pp.173-179.

[15] A.P.Fitz and R.J.Green, 1996, "Fingerprint Classification Using Hexagonal Fast Fourier Transform", Patter Recognition, Vol.29, No.10, pp. 1587-1597.
[16] B.G. Sherlock and D.M. Monoro, 1993, "A Model for Interpreting Fingerprint Topology", Pattern Recognition, Vol.26, No.7, pp.1047-1055.
[17] M.Kawagoe and A.Tojo, 1984, "Fingerprint Pattern Classification" Pattern Recognition Vol.17, No. 3, pp. 295-303.
[18] C.I Waston and C.L Wilson, 1992, "NIST Special Database 4, Fingerprint Database", National Institute of Science and Technology.
[19] M. Pontil, A. Verri, 2008, "Support Vector machines for 3-D Object recognition", IEEE Trans, Pattern Anal. Mach. Intell., Vol.20, No.6, pp.637-646.
[20] E.R Henry, 1900, "Classification and use of fingerprint" Routledge London.
[21] C.L Wilson, G.T Candela and C.I Watson, 2009 "Neural Network Fingerprint classification", J. Artificial Neural Network, Vol.1, No.12, pp.1-25.