JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Enhanced Offline Signature Recognition Using K-NN and SVM

Saravjeet Kaur
Computer Science & Engineering Department
Doaba Group of Colleges Kharar (Punjab) India
Moninder Pal Kaur Gill
Computer Science & Engineering Department
Doaba Group of Colleges Kharar (Punjab) India

ABSTRACT

Biometrics, a method of identifying individuals based on their physiological or behavioural traits, is adept at reliably distinguishing between authorized users and impostors. Signature verification systems fall into two categories: offline (static) and online (dynamic). This thesis introduces a recognition system for offline signatures, employing K-NN and SVM with SURF features. Notably, the system is trained on low-resolution scanned signature images. A person's signature serves as a crucial biometric attribute for authenticating identity. Despite being a human attribute, signatures can be treated as images and recognized using computer vision techniques, such as K-NN and SVM with SURF features. Given the capabilities of modern computers, the development of rapid algorithms for signature recognition becomes imperative. The field of signature recognition offers diverse approaches with ample research opportunities. This thesis focuses on offline signature recognition and verification through K-NN and SVM with SURF features. The proposed system captures signatures in image format and utilizes various image processing techniques to extract parameters for verification. Implementation is carried out using Matlab, and the effectiveness of this approach has been tested and confirmed. The utilization of Matlab underscores the practicality and suitability of this proposed methodology for offline signature recognition and verification.

Keywords- Offline signature recognition, K-NN, SVM

INTRODUCTION

We delve into the realm of enhanced offline signature recognition through the synergistic application of K-Nearest Neighbors (K-NN) and Support Vector Machines (SVM). Harnessing the power of these machine learning algorithms, our study aims to refine the accuracy and efficiency of recognizing handwritten signatures, thereby contributing to the advancement of biometric authentication systems. By analyzing the distinctive characteristics of signatures, we seek to optimize the error rejection rates and elevate the overall precision of the recognition process. This research represents a significant stride toward achieving more reliable and secure methods for verifying human identity through handwritten signatures.

OVERVIEW OF SIGNATURE RECOGNITION

A problem of personal verification and identification is an actively growing area of research. The methods are numerous and are based on different personal characteristics; voice, lip movement, hand geometry, face, odor, gait, iris, retina and fingerprint are the most commonly used authentication methods. All these psychological and behavioral characteristics are called biometrics. The driving force of the progress in this field is above all, the growing role of the internet and electronic transfers in modern society. Therefore considerable number of applications is concentrated in the area of electronic commerce and electronic banking systems.

The biometrics have a significant advantage over traditional authentication techniques (namely passwords, PIN numbers, smart cards etc) due to the fact that biometric characteristics of the individual are not easily transferable are unique of every person and cannot be lost, stolen or broken. The choice of one of the biometric solutions depends on several factors which include:

- User approval
- Security level prerequisites
- Precision and reliability
- Financial considerations and Implementation Timeline

The method of signature verification reviewed in this paper benefits the advantage of being highly accepted by potential customers. The use of the signature has a long history which

goes back to the appearance of writing itself. Utilization of the signature as an authentication method has already become a tradition in the western civilization and is respected among the others. The signature is an accepted proof of identity of the person in a transaction taken on his or her behalf. Thus the users are more likely to approve this kind of computerized authentication method. Signature verification systems differ in both their feature selection and their decision methodologies. More than 40 different feature types have been used for signature verification. Features can be classified into two major types: local and global. Global features are features related to the signature as a whole, for instance the average signing speed, the signature bounding box and Fourier descriptors of the signatures trajectory. Local features correspond to a specific sample point along the trajectory of the signature. Examples of local features include distance and curvature change between successive points on the signature trajectory. Most commonly used online signatures acquisition devices are pressure sensitive tablets capable of measuring forces exerted at the pen-tip, in addition to the coordinate of the pen. The pressure information at each point along the signature trajectory is another example of commonly used local feature. Some of these features are compared in order to find the more robust ones for signature verification purposes. Other systems have used genetic algorithms to find the most useful features. Due to the high sampling rate of the tablet, some consecutive sample points may mark the same trajectory point especially when the pen movement is slow. Most verification systems resample the input so as to obtain a trajectory consisting of equidistant points. This is often done in order to remove redundant points to speed up the comparisons and to obtain a shape-based representation, removing the time dependencies, separately keep track of the local velocity values and use them in aligning two signatures. Signature recognition and verification involves two separate but strongly related tasks: one of them is identification of the signature owner, and the other is the decision about whether the signature is genuine or forged. Also, depending on the need, signature recognition and verification problem is put into two major classes: (i) On-line signature recognition and verification systems (SRVS) and (ii) Off-line SRVS. On-line SRVS requires some special peripheral units for measuring hand speed and pressure on the human hand when it creates the signature. On the other hand, almost all Off-line SRVS systems rely on image processing and feature extraction techniques.

Image Preprocessing and Features Extraction

We approach the problem in two steps. Initially, the scanned signature image is preprocessed to be suitable for extracting features. Then, the preprocessed image is used to extract relevant geometric parameters that can distinguish forged signatures from exact ones using the ANN approach.

Preprocessing:

The signature is first captured and transformed into a format that can be processed by a computer. Now it's ready for preprocessing. In preprocessing stage, the RGB image of the signature is converted into grayscale and then to binary image. The purpose of this phase is to make signatures ready for feature extraction. The preprocessing stage includes two steps: Color inversion, Filtering and Binarization.

Color Inversion:

The true color image RGB is converted to the grayscale intensity image by eliminating the hue and saturation information while retaining the luminance.

Fig.1.(a) Asamplesignaturetobeprocessed;(b)AGrayscaleIntensityImage

A grayscale image is a data matrix whose values represent intensities within some range where each element of the matrix corresponds to one image pixel. Image Filtering and Binarization:

Any image when resample is filtered by a low pass FIR filter. This is done to avoid aliasing. This aliasing occurs because of sampling the data at a rate lower than twice the largest frequency component of the data. So a low pass filter will remove the image high frequency components. And for this purpose the filter used. Now the grayscale image is segmented to get a binary image of objects. In a binary image, each pixel assumes one of only two discrete values: 1 or 0. A binary image is stored as a logical array.

Fig.2.BinaryImageinterpretingthebitvalueof 0asblackand1as white

Features Extraction is the key to develop an offline signature recognition system. We use a set of five global features that cannot be affected by the temporal shift.

Types of Signature Verification

Based on the definitions of signature, it can lead to two different approaches of signature verification.

Off-Line or Static Signature Verification Technique

This approach is based on static characteristics of the signature which are invariant. In this sense signature verification, becomes a typical pattern recognition task knowing that variations in signature pattern are inevitable; the task of signature authentication can be narrowed to drawing the threshold of the range of genuine variation. In the offline signature verification techniques, images of the signatures written on a paper are obtained using a scanner or a camera.

On-line or Dynamic Signature Verification Technique

This is the second type of signature verification technique. This approach is based on dynamic characteristics of the process of signing. This verification uses signatures that are captured by pressure sensitive tablets that extract dynamic properties of a signature in addition to its shape. Dynamic features include the number of order of the strokes, the

Over all speed of the signature and the pen pressure at each point that make the signature more unique and more difficult to forge. Application areas of Online Signature Verification include protection of small personal devices (e.g. PDA, laptop), authorization of computer users for accessing sensitive data or programs and authentication of individuals for access to physical devices or buildings.

PROPOSED METHOD

i K-NN(K-Nearest Neighbors)

In pattern recognition,the *k*-Nearest Neighbors algorithm (or *K*-NN for short) is a non-parametricmethod used forclassificationand regression.In both cases, the input consists of the *k* closest training examples in the space. The output depends on whether *K*-NN is used for classification or regression:

- In *K-NN classification*, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer,typically small). If *k*=1, then the object is simply assigned to the class of that single nearest neighbor.

- In *K-NN regression*, the output is the property value for the object. This value is the average of the values of its *k* nearest neighbors.

K-NN is a type of instance-based learning,or lazy learning,where the function is only approximated locally and all computation is deferred until classification. The *k*-NN algorithm is among the simplest of all machine learningalgorithms.

Both for classification and regression, it can be useful to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of 1/*d*, where *d* is the distance to the neighbor.

The neighbors are taken from a set of objects for which the class(for *k*-NN classification) or the object property value (for *k*-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

Ii SUPPORT VECTOR MACHINE

The Support Vector Machine (SVM) stands as a cutting-edge classification method pioneered in 1992 by Boser, Guyon, and Vapnik. Widely embraced in bio informatics and various disciplines, the SVM classifier's popularity stems from its exceptional accuracy in handling high-dimensional data, such as gene expression, and its adaptability in modeling diverse datasets.

SVMs fall under the broader umbrella of kernel methods, a category of algorithms relying on dotproducts for computation. In this context, the dot product can be substituted with a kernel function, performing a dot product in a potentially high-dimensional feature space. This approach offers two notable advantages: firstly, the capacity to create non-linear decision boundaries, utilizing techniques originally designed for linear classifiers; and secondly, the ability to apply a classifier to data lacking an evident fixed-dimensional vector space representation. In bioinformatics, sequences like DNA or protein, as well as protein structures, exemplify such data.

Effectively harnessing SVMs necessitates a thorough understanding of their functioning. During SVM training, practitioners must make crucial decisions, including data preprocessing, kernel selection, and parameter tuning. Informed choices in these aspects are pivotal for optimal performance, as uninformed decisions may severely compromise results. The objective is to impart users with an intuitive grasp of these decision points and furnish general guidelines for usage. All the illustrative examples provided were generated using the PyML machine learning environment, which specifically emphasizes kernel methods and SVMs.

PRELIMINARIES: LINEAR CLASSIFIERS

Support vector machines serve as a prominent example of a linear two-class classifier, a concept explored in this section for better comprehension. In a two-class learning scenario, the dataset is comprised of objects labeled as +1 or -1. Here, boldface x denotes a vector with components xi, and xi represents the ith vector in the dataset denoted as f(xi, yi) for i=1 to n, where yi is the associated label. These objects, interchangeably referred to as patterns or examples, initially assume a vector format, but as kernels are introduced, this assumption becomes more flexible, allowing for any continuous/discrete object, such as a protein/DNA sequence or protein structure.

Central to the definition of a linear classifier is the concept of the dot product, also known as an inner product or scalar product. The weight vector (w) and bias (b) play pivotal roles in this context. In the scenario where b = 0, the set of points x satisfying wTx = 0 forms a hyperplane. Introducing the bias b facilitates the translation of the hyperplane away from the origin.

For b = 0, the hyperplane f(x): f(x) = wTx + b = 0 divides the space into two regions. The sign of the discriminant function f(x) indicates the side of the hyperplane a point occupies. The decision boundary of the classifier is the demarcation between regions classified as positive and negative. Described by a hyperplane, this decision boundary is referred to as linear, signifying its linearity in the input examples. Consequently, a classifier with a linear decision boundary is termed a linear classifier. Conversely, when the decision boundary exhibits a non-linear dependence on the data, the classifier is termed non-linear.

KERNELS: FROM LINEAR TO NON-LINEAR CLASSIFIERS

In numerous applications, the use of non-linear classifiers often yields superior accuracy. However, linear classifiers offer certain advantages, particularly in terms of simple training algorithms that scale efficiently with the number of examples [9, 10]. This prompts the question: Can the machinery of linear classifiers be extended to generate non-linear decision boundaries? Moreover, how can we address domains like protein sequences or structures where representation in a fixed-dimensional vector space is unavailable? The conventional approach to creating a non-linear classifier from a linear one involves mapping data from the input space X to a feature space F using a non-linear function.

However, explicitly computing non-linear features poses scalability challenges, especially when dealing with a large number of input features. For instance, applying the mapping in the above example leads to a quadratic increase in the dimensionality of the feature space F, resulting in a significant rise in memory usage and computation time for the discriminant function of the classifier. This quadratic complexity is manageable for low-dimensional data but becomes impractical for high-dimensional gene expression data.

Kernel methods offer a solution by bypassing the explicit mapping of data to a high-dimensional feature space. The Gaussian kernel, a widely used example, is defined by

Gaussian kernel is defined by:

Gaussian kernel is defined by:

$$K(\mathbf{x_i},\mathbf{x_j})=\exp(-\frac{\left\| \mathbf{x_i}-\mathbf{x} \right\|^2}{2\sigma^2}$$

Where > 0 is a parameter that controls the width of Gaussian. It plays a similar role as the degree of the polynomial kernel in controlling the exibility of the resulting classifier. We saw that a linear decision boundary can be kernelized i.e. its dependence on the data is only through dot products. In order for this to be useful, the training algorithms need to be kernelizable as well [6]. It turns out that a large number of machine learning algorithms can be expressed using kernels | including ridge regression, the perceptron algorithm, and SVMs.

In general, the RBF kernel is a reasonable first choice. This kernel nonlinearly maps samples into a higher dimensional space so it, unlike the linear kernel, can handle the case when the relation between class labels and attributes is nonlinear. Furthermore, the linear kernel is a special case of RBF since the linear kernel with a penalty parameter ~ C has the same performance as the RBF kernel with some parameters (C;). In addition, the sigmoid kernel behaves like RBF for certain parameters.

The polynomial kernel has more hyperparameters than the RBF kernel. There are some situations where the RBF kernel is not suitable. In particular, when the number of features is very large, one may just use the linear kernel.

CONCLUSION:

The basic advantage of implementing SVM and K-NN is that they can extract the most discriminative and representative of features. The proposed algorithm is implemented as a practical and helpful for signature verification and identification system. Algorithm proposed successfully made rotation invariant by using the rotation of the image. The error rejection rate(ERR) can further be improved by using enhanced techniques SVM and KNN.

REFERENCES

[1] Anil K Jain, Arun Ross and Salil Prabhakar, "An Introduction to Biometric Recognition,"*IEEETransactionsonCircuitsandSystemsforVideoTechnology*,vol.14, no.1, pp. 1-29, 2004.

[2] K.R. Radhika, M.K. Venkatesha and G.N. Sekhar, "Off-Line Signature AuthenticationBasedonMoment InvariantsUsingSupportVectorMachine",Journalof Computer Science 6 (3): 305-311, 2010.

[3] Reza Ebrahimpour, Ali Amiri, Masoom Nazari and Alireza Hajjany, "Robust Model for Signature Recognition Based on Biological Inspired Features", International Journal of Computer and Electrical Engineering, Vol. 2, No. 4, August, 2010.

[4]. A. Piyush Shanker, A.N. Rajagopalan, "Off-line signature verification using DTW", Pattern Recognition Letters 28 (2007) 1407–1414.

[5] AlessandroZimmerandLeeLuanLing, "OfflineSignatureVerificationSystemBased on the Online Data",EURASIP Journalon Advances in Signal Processing Volume 2008, Article ID 492910, 16 pages.

[6] BanshidharMajhi,YSanthoshReddy,DPrasannaBabu, "NovelFeaturesforOff-line

SignatureVerification" International Journal of Computers, Communications & Control,

Vol. I, No. 1, pp. 17-24, 2006.

[7] Migual A. Ferrer, Jesus B. Alonso and Carlos M. Travieso, "Off-line Geometric

Parametersfor Automatic SignatureVerification Using Fixed-Point Arithmetic", *IEEE Tran. on Pattern Analysis and Machine Intelligence*, vol.27, no.6, June 2005.

[8] Debasish Jena, Banshidhar Majhi, Saroj Kumar Panigrahy, Sanjay Kumar Jena "Improved

Offline Signature Verification Scheme Using Feature Point Extraction Method".

[9] B. Fang, C.H. Leung, Y.Y. Tanand Y.K. Wong, "Off-line signatures tracking of feature and stroke Recognition, pp. 91-101, 2003.

[10]G.A. Khuwaja. An adaptive combined classifier system for invariant face recognition. Digital Signal Processing 12, 21–46, DOI:10.1006/dspr.2001.0413, 2002. [11]D. Zhang, J. Campbell, D. Maltoni, and R. Bolle. Special issue on biometric systems. IEEE Trans. Systems, Man and Cybernetics - C, 35(3):273–275, 2005. [12]K. Bowyer, V.Govindaraju, andN.Ratha. Introductiontothespecialissueonrecent advances in biometric systems. IEEE Trans. Systems, Man and Cybernetics - B, 37(5): 1091–1095, 2007.

[13] G. A. Khuwaja. Fingerprint identification with LVQ. Proc of the 9th IEEE Int Conf on Neural Information Processing (ICONIP'OZ), Vol. 2, Singapore, Nov. 2002.

[14] J.F. Vargas, M.A. Ferrer, C.M. Travieso, and J. B. Alonso. Offline signature verification based on pseudo-cepstral coefficients. 10th IEEE Int Conf on Document Anal. & Recognition, 2009.

[15] A.I. Al-Shoshan. Handwritten signature verification using image invariants and dynamic features. Proc of the IEEE Int Conf on Computer Graphics, Imaging and Visualization (CGIV'06), 2006.

[16] M. Piekarczyk. Hierarchicalrandomgraph modelfor off-line handwrittensignatures recognition. IEEE Int Conf on Complex, Intelligent, Software Intensive Systems, 2010. [17]S.M.S. Ahmad, A. Shakil, M.A. Faudzi, R.M. Anwar. Analysis of 'goat' within user populationofanoffline signature biometrics. 10th IEEE Int ConfonInformationScience, Signal Processing and their Applications (ISSPA 2010).

[18] J.P. Drouhard, R. Sabourin, and M. Godbout. A neural network approach to off-line signature verification using directional PDF. Pattern Recognition, 29(3), (1996), 415-- 424.

[19] E.J.R.Justino,F.Bertolozzi,andR.Sabourin.AcomparisonofSVM and HMM classifiers in the offline signature verification. Pattern Recognition Letters, vol. 26, 1377-1385, 2005.

[20] K. Delac and M. Grgic. A survey of biometric recognition methods. Proc of 46th IEEE Int Symposium Electronics, Croatia, 184-193, June 2004.

[21] B. Kovari, Z. Kertesz, and A. Major. Off-line signature verification based on feature matching. 11th IEEE Int Conf on Intelligent Engineering Systems, Budapest, Hungary,29 June - 1 July 2007.

[22] A. Kholmatov and B. Yanikoglu. Identity authentication using improved on-line signature verification method. Pattern Recognition Letters, Volume 26, Issue 15, November 2005, pp. 2400-2408.

[23] T.S. Ong, W.H. Khoh, A. Teoh. Dynamic handwritten signature verification based on statistical quantization Mechanism. IEEE Int Conf on Computer Engineering and Technology, 2009.

[24] J. Edson, R. Justino, F. Bortolozzi, and R. Sabourin: "An off-line signature verification using HMM for random, simple and skilled forgeries," Proc. Sixth Int. Conf. Document Analysis and Recognition, pp. 1031-1034, Sept. 2001.

[25] B. Zhang, M. Fu and H. Yan: "Handwritten signature verification based on neural

'Gas' based vector quantization", IEEE Int.Joint Conf. on Neural Networks, pp. 1862- 1864, May1998. verification using compression networks and positionalcuttings", Proc. 2003 IEEE Workshop on Neural Networks for Signal Processing, vol. 1, pp. 627-636, 2003.

[26] A. Chalechale and A. Mertins: "Line segment distribution of sketches for persian

signature recognition", IEEE Proc. TENCON, vol. 1, pp. 11–15, Oct. 2003.