An Efficient Word Recognition System Based on Effective Classification Method

[1]P. B. KUMBHAR
T.P.C.T.'s College Of Engineering, Osmanabad.
Maharashtra, India.
[2]A. N. HOLAMBE
T.P.C.T.'s College Of Engineering, Osmanabad.
Maharashtra, India.

Abstract

Word spotting is a technique which can extract the text from input image. This paper addresses the problems of word spotting and word recognition on images. . In recognition, the goal is to recognize the content of the word image, usually aided by a dictionary or lexicon. Here we describe an approach in which both word images and text strings are embedded in a common vectorial subspace. This is achieved by a combination of label embedding and attributes learning, and a common subspace regression. In this subspace, images and strings that represent the same word are close together, allowing one to cast recognition and retrieval tasks as a nearest neighbor problem. In this work we shall also assume that the location of the words in the images is provided, i.e., we have access to images of cropped words. If those were not available, text localization and segmentation techniques could be used. In this paper the word can be recognized through this process using different algorithms.

Keywords*: word spotting, recognition, text, feature, k nearest neighbor.*

1. INTRODUCTION

An image is an array, or a matrix, of square pixels (picture elements) arranged in columns and rows. Pictures are the most common and convenient means of conveying or transmitting information. A picture is worth a thousand words. Pictures concisely convey information about positions, sizes and inter relationships between objects. They portray spatial information that we can recognize as objects. Human beings are good at deriving information from such images, because of our innate visual and mental abilities. About 75% of the information received by human is in pictorial form.

An image is digitized to convert it to a form which can be stored in a computer's memory or on some form of storage media such as a hard disk or CD-ROM. This digitization procedure can be done by a scanner, or by a video camera connected to a frame grabber board in a computer. Once the image has been digitized, it can be operated upon by various image processing operations. Image processing operations can be roughly divided into three major categories, Image Compression, Image Enhancement and Restoration, and Measurement Extraction. The problem of multi-writer word spotting can be removed in this paper [1]. The objective of word spotting is to find all instances of a given word in a potentially large dataset of document images. This is typically done in a query-by-example (QBE) scenario, where the query is an image of a handwritten word and it is assumed that the transcription of the dataset and the query word is not available.[6]Author present a word spotting system based on Hidden Markov Models (HMM) that uses trained sub word models to spot keywords. The standard technique for indexing documents is to scan them in, convert them to machine readable form (ASCII) using Optical Character Recognition (OCR) and then index them using a text retrieval engine. However, OCR does not work well on handwriting. Here an alternative scheme is proposed in this paper for indexing such texts. Each page of the document is segmented into words. The images of the words are then matched against each other to create equivalence classes. The user then provides ASCII equivalents for say the top 2000 equivalence classes [8]. In this paper [20] introduces a product quantization based approach for approximate nearest neighbor search. The idea is to decompose the space into a Cartesian product of low dimensional subspaces and to quantize each subspace separately. A vector is represented by a short code composed of its

subspace quantization indices. The Euclidean distance between two vectors can be efficiently estimated from their codes. An asymmetric version increases precision, as it computes the approximate distance between a vector and a code. In this paper we describe an approach in which both word images and text strings are embedded in a common vectorial subspace. This is achieved by a combination of label embedding and attributes learning, and a common subspace regression. In this subspace, images and strings that represent the same word are close together, allowing one to cast recognition and retrieval tasks as a nearest neighbor problem. In this work we will also assume that the location of the words in the images is provided, i.e., we have access to images of cropped words. If those were not available, text localization and segmentation techniques could be used. This paper is discussed in following section. Section 2 deals with Related work. Section 3 Algorithms, Section 4 deals with Result and conclusion and section 5 deals with References.

2. RELATED WORK

Here we review the works most related to some key aspects of our desired work.

2.1 Word Spotting and Recognition in Document Images

TEXT understanding in images is an important problem that has drawn a lot of attention from the computer vision community since its beginnings. Text understanding covers many applications and tasks, most of which originated decades ago due to the digitalization of large collections of documents. This made necessary the development of methods able to extract information from these document images: layout analysis, information flow, transcription and localization of words, etc. Recently, and motivated by the exponential increase of publicly available image databases and personal collections of pictures, this interest now also embraces text understanding on natural images. Methods able to retrieve images containing a given word or to recognize words in a picture have also become feasible and useful. In this paper we consider two problems related to text understanding: word spotting and word recognition. In word spotting, the goal is to find all instances of a query word in a dataset of images. The query word may be a text string – in which case it is usually referred to as query by string (QBS) or query by text (QBT), or may also be an image, in which case it is usually referred to as query by example (QBE). In word recognition, the goal is to obtain a transcription of the query word image. In many cases, including this work, it is assumed that a text. Dictionary or lexicon is supplied at test time, and that only words from that lexicon can be used as candidate transcriptions in the recognition task. In this work we will also assume that the location of the words in the images is provided, i.e., we have access to images of cropped words.

2.2 Block Diagram

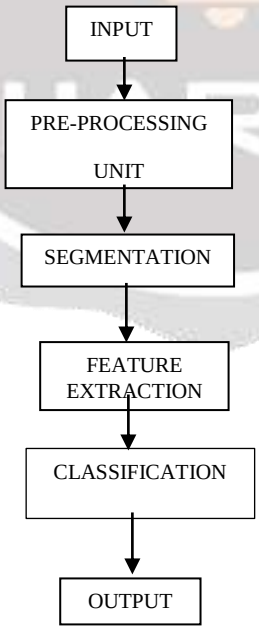

Fig. 1 Block diagram of Word spotting

2.2.1 Dataset Preparation

We collect the Tamil land documents from friends and neighbor nearly up to 80 documents. The documents were real and accurate would not be faked one. We can manually cropped the each word for better accuracy .we can cropped the word from document it reaches up to 800 words .we kept the dataset separately for preprocessing stages.

2.2.2 Preprocessing

Preprocessing is a technique which would remove the noise from the input image using such technique like binarization, skew correction and slant correction etc., Image Enhancement is a method which improves the quality of images for different formats like .gif, .tiff., .jpg etc., Binarization is a method carries a logical value like zero (0) and ones (1).It takes a logical value ones (1) for foreground and zeros (0) for background. It stores on logical array. Skew correction and slant correction method are used for properly aligned the page layout for misaligned document.

2.2.3 Segmentation

Segmenting a page into paragraph, paragraph into lines and lines into word. In this work we will also assume that the location of the words in the images is provided, i.e., we have access to images of cropped words. If those were not available, text localization and segmentation techniques could be used .In the segmentation process we are cropping the words identically and show it in the bounding box.

2.2.4 Feature extraction

It is the heart of a pattern recognition application. Feature extraction techniques like Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Gradient based Feature Zoning and Euclidean distance. Histogram might be applied to extract the feature of individual characters. This stage is represented as a feature vector which it is identity one. The main goal is extract a set of features from recognized character. It improves the recognition rate and reduces the misclassification. Here, we used Gabor feature who calculated the statistical region properties for each word. Gabor filter is a linear filter and it widely used in pattern recognition for extracting the feature of an image. The Gabor space is used in image processing application on such as optical character recognition, iris recognition and fingerprint recognition. It has a specific spatial location are very distinctive between objects in an image. It can be extracted from the Gabor space in order to create a sparse object representation.

2.2.5 Classification

Classification is a technique which classifies a data from training dataset which compared to unknown Labels and finally created a new one. Classification has two process are model usage and model construction method. Model usage method deals with classifying the previously unseen objects. The test sample is compared with classified data for getting accuracy rate of classifier. Test set is independent of trained set which avoid pruning. Model construction method deals with previously known objects.

2.3 Flow Diagram

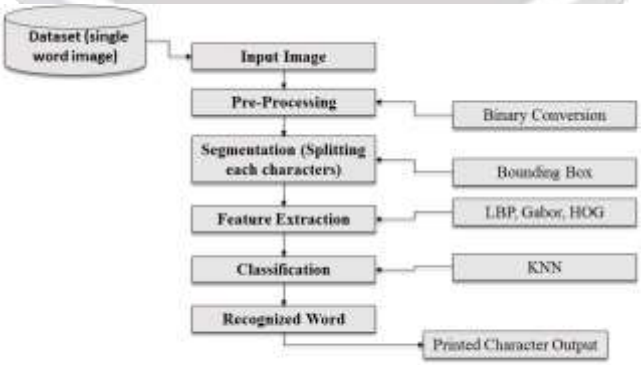

Fig. 2 System Architecture of Word Spotting Process

3. ALGORITHMS

3.1 LBP:

Local binary patterns (LBP) are a type of visual descriptor used for classification in computer vision. The local binary pattern operator is an image operator, which transforms an image into an array or image of integer labels describing small-scale appearance of the image [19]. It has proven to be highly discriminative and its key points of interest, namely its invariance to monotonic gray level changes and computational proficiency, make it suitable for demanding image analysis tasks. The basic local binary pattern operator, introduced by Ojala et. al. [20], was based on the assumption that texture has locally two complementary aspects, a pattern and its strength. LBP feature extraction consists of two principal steps: the LBP transform, and the pooling of LBP into histogram representation of an image. As explained in [20] gray scale invariance is achieved because of the difference of the intensity of the neighboring pixel to that of the central pixel. It also encapsulates the local geometry at each pixel by encoding binarized differences with pixels of its local neighborhood:

$$LBP_{P,R,t} = \sum_{P=0}^{P-1} s_t(gp - gc) \times 2^p,$$

Where gc is the central pixel being encoded, gp are P symmetrically and uniformly sampled points on the periphery of the circular area of radius R around gc, and st is a binarization function parameter by t. The sampling of gp is performed with bilinear interpolation. t, which in the standard definition is considered zero, is a parameter that determines when local differences are considered big enough for consideration. In our LBP the original version of the local binary pattern operator works in a 3*3 pixel block of an image. The pixels in this block are threshold by its center pixel value, multiplied by powers of two and then summed to obtain a label for the center pixel. As the neighborhood consists of 8 pixels, a total of 28 = 256 different labels can be obtained depending on the relative gray values of the center and the pixels in the neighborhood. In our case, we use the uniform LBP as mentioned in Ojala et.al. [20] which are the fundamental property of LBP, for the development of a generalized gray scale invariant operator.

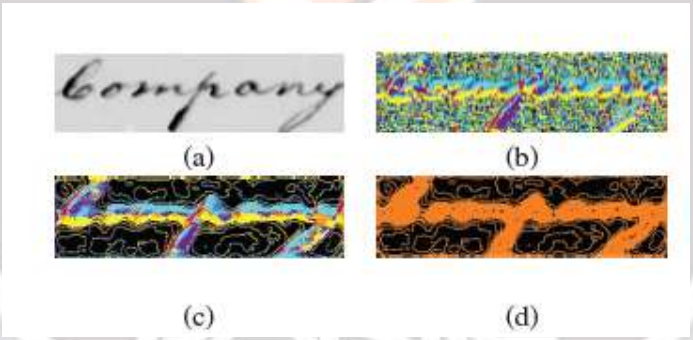

Fig. 3 (a) Input Image (b) LBP image (c) LBP with median filtering (d) LBP Uniformity with median filter.

Fig 3 (a). The term 'uniform' in case of local binary pattern refers to the uniformity of the appearance i.e. the circular presentation of the pattern has a limited number of transitions or discontinuities. The patterns which are considered as uniform provide a vast majority over 90%, of the 3*3 texture patterns in the historical documents. The most frequently observed 'uniform' patterns correspond to fundamental micro-features such as corners, spot and edges as shown in Fig 3 (b). These are also considered as feature detectors for triggering the best pattern matching. In our case, where P = 8, $LBP_{8,R}$ can have 256 different values. We also designate patterns that have uniformity value of at most 2 as 'uniform' and uses the following Eq

$$LBP_{P,R}^{u^2} = \begin{cases} \sum_{p=0}^{p-1} s(g_p - g_c), & if\ U(LBP_{P,R}) \le 2 \\ P+1, & otherwise \end{cases}$$

Where

$$U(LBP_{P,R}) = |s(g_{p-1} - g_c) - s(g_o - g_c)|$$

$$+\sum_{P=1}^{P-1} \left| s(g_p - g_c) - s(g_{p-1} - g_c) \right|$$

The U value is the uniformity value for at most 2. From definition exactly P + 1 'uniform' binary patterns can occur in a circularly symmetric neighbour set of P pixels. Eq. 3 assigns a unique label to each of them, corresponding to the number of '1' bits in the pattern (0-> P), while the 'no uniform' patterns are grouped under the miscellaneous label (P+1). The final texture feature employed in texture analysis is the histogram of the operator outputs (i.e. pattern labels) accumulated over a texture sample. The reason why the histogram of 'uniform' patterns provides better discrimination in comparison to the histogram of all individual patterns comes down to differences in their statistical properties Fig. 3(d). The relative proportion of 'no uniform' patterns of all patterns accumulated into a histogram is so small that their probabilities cannot be estimated reliably.

3.2 HOG:

The histogram of oriented gradients (HOG) is a feature descriptor used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms, scale-invariant feature transform descriptors, and shape contexts, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.

3.3 GABOR:

In image processing, a Gabor filter, named after Dennis Gabor, is a linear filter used for edge detection. Frequency and orientation representations of Gabor filters are similar to those of the human visual system, and they have been found to be particularly appropriate for texture representation and discrimination. In the spatial domain, a 2D Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave.

3.4 KNN:

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

4. RESULT ANALYSIS AND CONCLUSION

Word spotting is a technique which implemented a pre-processing technique like noise removal and post processing technique like classification and recognition. We get an accurate recognized character from the input image using MATLAB and hence, we processed the images. Thus we describe an approach in which both word images and text strings are embedded in a common vectorial subspace. This is achieved by a combination of label embedding and attributes learning, and a common subspace regression. In this work we shall also assume that the location of the words in the images is provided, i.e., we have access to images of cropped words. Thus, the word can be recognized through this process using different algorithms.

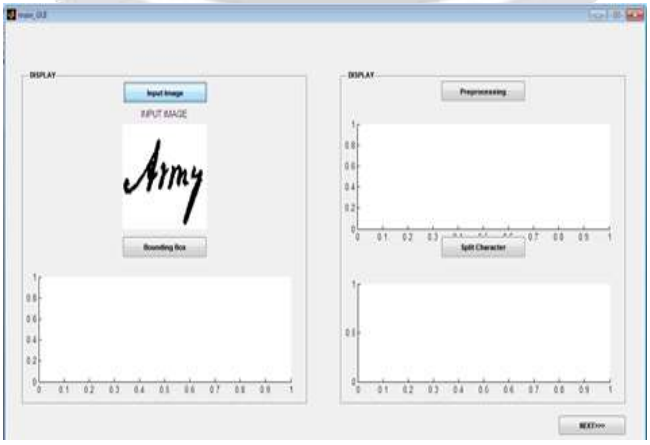

Fig 4: Input image

Fig 5: Binary Image

Fig 6: Bounding Box Segmentation

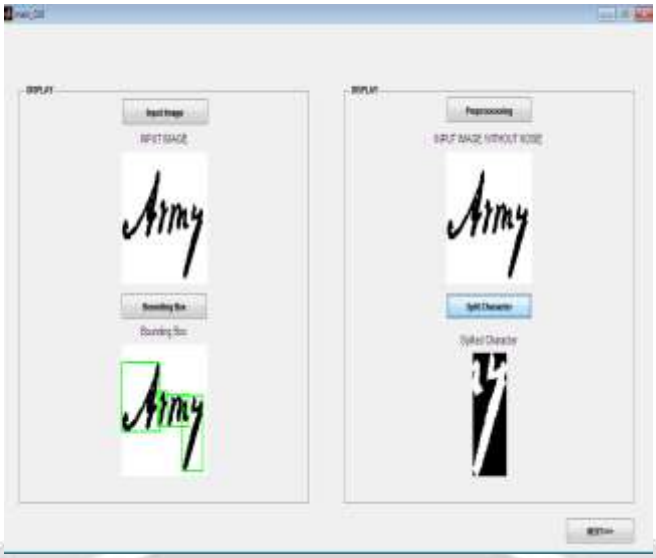

Fig 7: Segmenting Letters

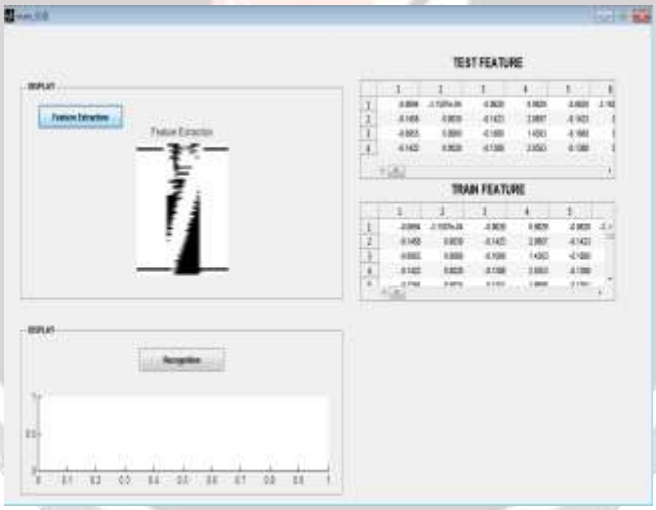

Fig 8: Feature Extraction

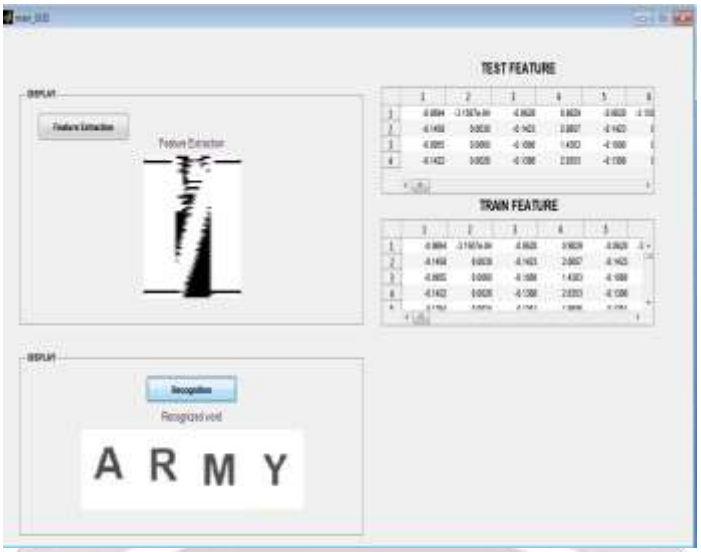

Fig 9: Recognized word using classifier

5. REFERENCES

[1] J. Almazan, A. Gordo, A. Fornes, and E. Valveny, "*Handwritten word spotting with corrected attributes*," in ICCV, 2013.

[2] R. Manmatha and J. Rothfeder, "*A scale space approach for automatically segmenting words from historical handwritten documents*," IEEE TPAMI, 2005.

[3] L. Neumann and J. Matas, "*Real-Time Scene Text Localization and Recognition*," in CVPR, 2012.

[4] L. Neumann and J. Matas, "*Scene Text Localization and Recognition with Oriented Stroke Detection*," in ICCV, 2013.

[5] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, "*PhotoOCR: Reading Text in Uncontrolled Conditions*," in ICCV, 2013.

[6] A. Fischer, A. Keller, V. Frinken, and H. Bunke, "*HMM-based word spotting in handwritten documents using subword models*," in ICPR, 2010.

[7] V. Frinken, A. Fischer, R. Manmatha, and H. Bunke, "*A novel word spotting method based on recurrent neural networks*," IEEE TPAMI, 2012.

[8] R. Manmatha, C. Han, and E. M. Riseman, "*Word spotting: A new approach to indexing handwriting*," in CVPR, 1996.

[9] T. Rath, R. Manmatha, and V. Lavrenko, "*A search engine for historical manuscript images," in SIGIR*, 2004.

[10] T. Rath and R. Manmatha, "*Word spotting for historical documents*," IJDAR, 2007.

[11] J. A. Rodr´ıguez-Serrano and F. Perronnin, "*Local gradient histogram features for word spotting in unconstrained handwritten documents*," in ICFHR, 2008.

[12] J. A. Rodr´ıguez-Serrano and F. Perronnin, "*A model-based sequence similarity with application to handwritten wordspotting*," IEEE TPAMI, 2012.

[13] S. Espana-Bosquera, M. Castro-Bleda, J. Gorbe-Moya, and ˜ F. Zamora-Martinez, "*Improving offline handwritten text recognition with hybrid HMM/ANN models*," IEEE TPAMI, 2011.

[14] I. Yalniz and R. Manmatha, "*An efficient framework for searching text in noisy documents*," in DAS, 2012.

[15] K. Wang, B. Babenko, and S. Belongie, "*End-to-end Scene Text Recognition*," in ICCV, 2011. [16] A. Mishra, K. Alahari, and C. V. Jawahar, "*Top-down and bottom-up cues for scene text recognition*," in CVPR, 2012.

[17] J. A. Rodr´ıguez-Serrano and F. Perronnin, "*Label embedding for text recognition*," in BMVC, 2013.

[18] C. Leslie, E. Eskin, and W. Noble, "*The spectrum kernel: A string kernel for SVM protein classification*," in Pacific Symposium on Biocomputing, 2002.

[19] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "*Text classification using string kernels*," JMLR, 2002.

[20] H. Jegou, M. Douze, and C. Schmid, "*Product quantization for ´ nearest neighbor search*," IEEE TPAMI, 2011.

[21] S. Lu, L. Li, and C. L. Tan, "*Document image retrieval through word shape coding*," IEEE TPAMI, 2008.

[22] P. Keaton, H. Greenspan, and R. Goodman, "*Keyword spotting for cursive document retrieval*," in DIA, 1997.

[23] F. Perronnin and J. A. Rodr´ıguez-Serrano, "*Fisher kernels for handwritten word-spotting*," in ICDAR, 2009.

[24] T. Jaakkola and D. Haussler, "*Exploiting generative models in discriminative classifiers*," in NIPS, 1999.

[25] B. Gatos and I. Pratikakis, "*Segmentation-free word spotting in historical printed documents*," in ICDAR, 2009.

[26] M. Rusinol, D. Aldavert, R. Toledo, and J. Llad ˜ os, "*Browsing ´ heterogeneous document collections by a segmentation-free word spotting method*," in ICDAR, 2011.

[27] J. Almazan, A. Gordo, A. Forn ´ es, and E. Valveny, "*Efficient ´ exemplar word spotting*," in BMVC, 2012.

[28] N. Dalal and B. Triggs, "*Histograms of oriented gradients for human detection*," in CVPR, 2005.

[29] T. Malisiewicz, A. Gupta, and A. Efros, "*Ensemble of Exemplar-SVMs for object detection and beyond*," in International Conference on Computer Vision, 2011, 2011.

[30] A. Mishra, K. Alahari, and C. V. Jawahar, "*Scene text recognition using higher order language priors*," in BMVC, 2012.

[31] A. Mishra, K. Alahari, and C. V. Jawahar, "*Image Retrieval using Textual Cu es*," in ICCV, 2013.