DETECTING TEXT IN IMAGES AND VIDEO FRAMES USING POLYNOMIAL KERNEL

K. Ilakkia Tamil
(Department of CSE, P. A. College of Engineering and Technology, Coimbatore, Tamil Nadu, tamililakkia93@gmail.com)

Abstract— *Text embedded in images and a video frame provides brief and important information about the content that can be used for indexing and retrieving of images and videos from large web databases efficiently. A coarse to fine algorithm is used to detect text lines in images and video frames under complex background. Coarse detection obtains the candidate text regions using the property dense intensity variety of text regions and contrast between text and its background by employing wavelet decomposition and density based region growing methods. Fine detection uses the texture property to discriminate between text and non-text pattern, it is done by employing four feature extractions wavelet moment feature, wavelet histogram feature, wavelet co-occurrence feature and crossing count histogram feature. Before classification the effective features from extracted features are selected using forward selection algorithm. Finally to detect text from non-text with polynomial kernel function Support Vector Machine (SVM) classifier is used.*

Keywords— Density based region growing; Feature Extraction; Feature selection; SVM classification; Text detection; Wavelet decomposition.

1. INTRODUCTION

Multimedia information in web databases is increasing. It is difficult task to manage and retrieve this information effectively. Text in images and video frames has high level semantic information that can be used for indexing and retrieving of multimedia components. In news videos caption text annotates information of the happening events [6,17]. In sport videos subtitle annotates information of score, athlete and highlight [14,16]. The huge amounts of data are carried by images and videos it is important to detect and identify text region as accurately as possible [10].

Text detection is about finding region in the image that contains the text. Text detection is a difficult task because text may be inserted into complex background [1], discrimination of text from other text like things, such as windows curtains, leaves or other general texture is difficult as it needs effective feature extraction technique, text in images has different font size, font color and language [5,8,9]. Furthermore the image digitalization and compression may introduce noise that may blur the embedded text characters. Our coarse to fine detection algorithm detects text in images and video frames under complex background. Different text properties are applied in various stages of detection. In coarse detection candidate text regions are obtained using the property dense intensity variety and contrast between text and its background. Structural information is used to separate text regions into text lines. In fine detection texture property is used to discriminate text from non text. Four kinds of texture features are extracted to identify candidate text lines. Figure 1 is the flow chart of coarse to fine. Feature selection algorithm is used to find effective features for classification. SVM classifier [3] with polynomial kernel is used to classify text from non text.

Advantages of proposed method are detection is made faster by using feature selection algorithm which reduces the features used for classification. Multiscale detection uses wavelet feature to detect text of different font size in image without scaling down operation of the original image. Four feature extractions are employed to find the texture feature in an image.

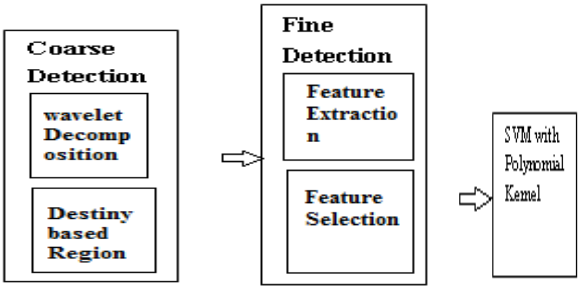


Fig.1 Flow Chart of Coarse to Fine

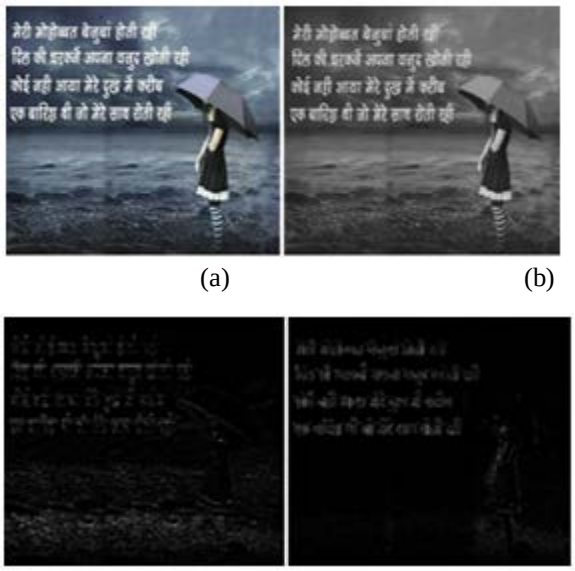
Advantages of proposed method are detection is made faster by using feature selection algorithm which reduces the features used for classification. Multiscale detection uses wavelet feature to detect text of different font size in image without scaling down operation of the original image. Four feature extractions are employed to find the texture feature in an image.

2. COARSE DETECTION

Candidate text lines are found in coarse detection. Multiscale wavelet decomposition is used to find text of different font size and candidate text pixels. Region growing method is employed to connect text pixels into text regions [15]. Obtained text regions are separated into candidate text lines using texture property.

2.1 Daubichie Wavelet Decomposition

Daubichie4 wavelet transformation is used for Multiscale wavelet decomposition as it has good location performance [2,13]. It is applied using Equation 1,



(a) (b)

((c) (d)

Fig.2 Two level wavelet decomposition. (a) Original image (b) Approximation image (c) Vertical image (d) Horizontal image

$$I_n(b_i, b_j) = [G_x * [G_y * I_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2}(b_i, b_j) \quad (1)$$

$$D_{n1}(b_i, b_j) = [H_x * [G_y * I_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2}(b_i, b_j)$$

$$D_{n2}(b_i, b_j) = [G_x * [H_y * I_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2}(b_i, b_j)$$

$$D_{n3}(b_i, b_j) = [H_x * [H_y * I_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2}(b_i, b_j)$$

Where I_0 is original image, * denotes convolution operator, H and G are high pass and low band pass filters, ($_{\downarrow 2,1}$) sub sampling along rows, D_{nk} are the wavelet coefficient with intensity variety information at level n and b_i and b_j are the locations in decomposition level. The above figure 2 shows the two level of wavelet decomposition. Text of different size can be detected in different decomposition levels.

Candidate text pixels are detected using the dense intensity variety around the text pixels. The wavelet energy feature of a pixel is calculated in Equation 2,

$$E_n(b_i, b_j) = (\sum_{k=1}^{3} [D_{nk}(b_i, b_j)]^2)^{1/2} \quad (2)$$

WHERE E_N IS THE ENERGY OF PIXEL AT LOCATION (I, J) AT LEVEL N IS SHOWS IN FIGURE 3.

Fig.3. Energy Level of an Image

2.2 Density Region Growing Method

Text regions are formed by connecting the text pixels. In morphological close operation all the pixels close to each other are connected to form a cluster of text pixels. The below figure 4 shows the candidate text regions. In density based region growing method seed pixel S is selected, if the percentage of pixels in its neighborhood is larger than the threshold T_D which is set 0.1. A pixel S´ is density connected with pixel S when S´ is within seed pixel neighborhood S. The region growing method as follows,

1. Select a seed pixel from the unlabeled candidate text pixels.
2. A new region is created if a seed pixels S is found, then unlabeled candidate pixels are iteratively collected that are density connected with S and labeled with same region label.
3. If there are more seed pixels still present go to (1)
4. Text regions formed are labeled. The pixels that are not included in the region are merged into with the background.

Fig.4 Candidate Text Region (a) Original Image (b) Candidate Text Pixel in First Scale (c) Horizontal Candidate region by Close Operation (d) Text region Detection by Density Based Region Growing

2.3 Separating Text Lines in Text Regions

Text regions consist of multiple text lines. Structural property is used to separate text lines from the detected text regions. The horizontal profile projection method is employed which calculates the sum of the candidate pixels of rows to separate the text lines is described in Equation 3,

$$T_p = (Avg_{profile} + Min_{profile})/2.0 \quad (3)$$

Where T_p is the threshold for finding the profile valley to separate the lines. $Min_{profile}$ and $Avg_{profile}$ the minimum and average profile values.

3. FINE DETECTION

Texture features are used to identify text from other text like features such as window curtains leaves etc [12]. Four types of feature extractions are used in this algorithm to identify candidate text.

3.1 Feature Extraction

Texture properties are used to identify text in an image. Regularity and directionality texture properties of a text in an image are weak [4,15]; therefore one kind of texture feature is insufficient for identifying text. Four kinds of features extractions methods are used to represent a text line. Three feature extractions are done in wavelet domain and one in gradient image.

Text of small font size is considered to be at high frequency signal while text of large font size is at low frequency signal [7, 11]. Features are extracted in the level where the candidate text lines are at suitable scale.

3.1.1 Wavelet Moment Feature

The wavelet mean and central features are extracted to represent the different intensity variance and spatial grey value distribution of text and non text. These features are texture property in wavelet domain. The features are calculated in Equation 4,

$$m(T) = \frac{1}{M*N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} T(i,j) \quad (4)$$

$$\mu_2(T) = \frac{1}{M*N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}(T(i,j) - m(T))^2$$

$$\mu_3(T) = \frac{1}{M*N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}(T(i,j) - m(T))^3$$

where T is a text line of size MxN, M is the mean, μ_2 is a second order central moment, μ_3 is the third order central moment, $T(i,j)$ wavelet coefficient of pixel (i, j). Features are extracted in three sub bands (LH, HL and HH) of high frequency.

3.1.2 Wavelet Histogram Feature

The energy distribution of a text line is represented using Wavelet Energy Histogram (WEH). Wavelet energy of all pixels is quantized into 16 levels to calculate the WEH(i).Quantization is done by Equation 5,

$$WE_q = WE \times 16/(WE_{max} - WE_{min}) \quad (5)$$

Where WE is a wavelet energy of a pixel, WE_{max} and WE_{min} are maximum and minimum energy value of image. The WEH(i) value is the percentage of pixels whose quantized energy is equal to i.

3.1.3 Wavelet Co-occurrence Feature

It is a second order statistic describing correlation among adjacent pixels. This improves the discrimination power between text and non text. The co-occurrence features are calculated in Equation 6,

$$E(d,\theta) = \sum_{i,j} c^2(d,\theta) \quad (6)$$

$$H(d,\theta) = \sum_{i,j} C(d,\theta)\, log\, C(d,\theta)$$

$$I(d,\theta) = \sum_{i,j}(i-j)^2\, C(d,\theta)$$

$$L(d,\theta) = \sum_{i,j}\frac{1}{1+(i-j)^2} C(d,\theta)$$

$$C(d,\theta) = \frac{\sum_{i,j}(i-\mu_x)(j-\mu_y)C(d,\theta)}{\sigma_x\sigma_y}$$

Where E (d, θ) is energy, C (d, θ) is co-occurrence matrix of element (i, j) with i co-occurring with j at a distance d in the direction θ. μ_x, μ_y and σ_x, σ_y are means and variances of the concurrence matrix $C(d,\theta)$. θ is selected as 0, 45, 90 and 135 degree. d is the co-occurrence distance set to be 1, 3 and 5 pixels. The features are calculated in 12 co-occurrence matrix in 3 wavelet sub bands.

3.1.4 Gradient Crossing Count Histogram Feature

Normalized Crossing Count Histogram (CCH) on Gradient Projection Map (GPM) is used to capture the periodicity of text along the text line. The crossing count of horizontal scan line k and N as the scan line number is calculated using Equation 7,

$$CCH'(K) = \frac{CC(K)}{\sum_{i=1}^{N} CC(i)} \quad (7)$$

The CCH normalized into 16 bins described in Equation 8,

$$CCH(i) = \frac{1}{16}\sum_{k=i\frac{N}{16}}^{(i+1)\frac{N}{16}} CCH'(k) \quad (8)$$

Where (i(N/16), (i+1)(N/16) represents a non overlapped window.

Table 1 Feature used for classification

Feature Set	Feature Description	Number of Features	Selected Features
Wavelet Moment Feature	Mean, Second and Third order moments	9	6
Wavelet Histogram Feature	Energy Histogram	16	9
Wavelet Co-occurrence Feature	Energy, Entropy, Homogeneity and Correlation	175	15
Gradient Crossing Count Histogram Feature	Crossing Count Histogram	16	9
Total		216	39

3.2 Feature Selection

All of the extracted features can be used to distinguish text with nontext; some features may contain

more information than others. Using all the extracted features for classification is time consuming so only a small set of the most powerful features are selected to reduce the time for feature extraction and classification. A forward search algorithm is used for selecting the effective features for SVM classifier.

The feature set E is first divided into selected feature set E_S and unselected feature set E_U, and then selected one by one using the following procedure:

1. Set E_S=ɵ and E_U=E
2. Label all of the features in E_U untested
3. Select one untested feature f from E_U and label it as tested
4. Put f and E_S together to form the temporary testing feature set F_S
5. Evaluate the classification performance of F_S; Accuracy is defined in Equation 9,

Accuracy= Number of correctly classified samples/ Number of samples (9)

6. If there are still untested features in E_U, goto (3);
7. Find a feature f if added into th e feature set F_S, the highest classification accuracy will be obtained f =argmax Accuracy (F_S) and then move f from E_U to E_S;
8. If there are still untested features in E_U, goto (2) and if E_U is empty, the procedure exists.

3.3 Classification

Support Vector Machine (SVM) classifier have learning algorithms as supervised learning models to analyze data and recognize patterns for classification [8]. The aim of a Support Vector Machine is to computes efficient separating hyperplanes in a high dimensional feature space. An optimal boundary between the possible outputs are found based on the transformation of data is called as kernel trick. The linear SVM can be extended to a nonlinear classifier by first using the nonlinear operator to map the input pattern into a higher dimensional space.

The polynomial kernel is a non linear kernel function commonly used with SVM and other kernelized models. It represents the similarity of vectors in a feature space over polynomials of the original variables. The kernel function in a SVM implicitly maps the input vector into high dimensional feature space. The polynomial kernel function assumes its maximum, x and y are aligned in the same direction that described in Equation 10,

$$K(x,y)=(x^T y+1)^p \quad (10)$$

Where p>0 is a constant that defines the kernel order. The optimal separating hyperplane is created and supported by support vector know as training points. The optimized hyperplane separates to two different cases with maximum distance.

4. RESULTS

This algorithm is applied for complex images with text and videos containing text. Texts are detected as shown in figure 5.

(a)

(b)

Fig.5. Text Detection Result

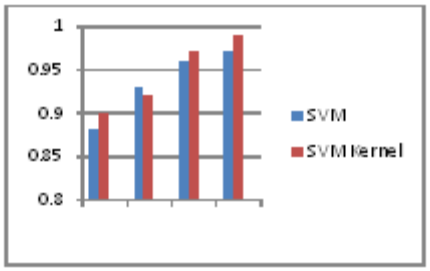


Fig.6. Comparison of classification performance

5. CONCLUSION

The coarse to fine algorithm used detect the text in image and video frames of different font size, color and language robustly. This method used to select the text lines instead of text regions which can be used for recognition using Optical Character Recognition (OCR) in future work. The combination of texture property for classification detects text faster. SVM classification has great generalization ability so it requires fewer training samples which is got by feature selection algorithm. Using of non linear polynomial kernel function in SVM makes classification efficient. The use of polynomial kernel has reduced the false alarm rate.

REFERENCES

[1] Chen D, Odobez J. M and Bourlard H (2002), 'Text segmentation and recognition in complex background based on Markov random field', Proceedings of the International Conference on Pattern Recognition, Mumbai, pp. 227–230.

[2] Ye Q, Huang Q, Gao W and Zhao D (2005), 'Fast and robust text detection in images and video frames', Image and vision computing, Beijing, China, pp.565-576.

[3] Chen D.T, Bourlard H and Thiran J. P (2001), 'Text identification in complex background using SVM', International Conference on Computer Vision and Pattern Recognition, Switzerland, pp. 621–626.

[4] Heisele B, Serre T, Mukherjee S and Paggio T (2001), 'Feature reduction and hierarchy of classifiers for fast object detection in video images', International Conference on Computer Vision and Pattern Recognition, Hawaii, pp. 18–24.

[5] Hua X.S, Liu W.Y and Zhang H.J (2004), 'An automatic performance evaluation protocol for video text detection algorithms', IEEE Transactions on Circuits and Systems for Video Technology, Seattle, volume 14, pp. 498–507.
[6] Hua X.S, Yin P and Zhang H (2002), 'Efficient video text recognition using multiple frame integration', International Conference on Image Processing, New York, pp. 22–25.
[7] Jain A.K and Yu B (1998), 'Automatic text location in images and video frames', Pattern Recognition 31, USA, pp. 2055–2076.
[8] Kim K.I, Jung K and Kim H (2003), 'Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm', IEEE Transactions on PAMI 25, USA, pp. 1631–1639.
[9] Li H and Doermann D (1999), 'Text enhancement in digital video using multiple frame integration', ACM Multimedia, College Park, pp. 385–395.
[10] Li H, Doermann D and Kia O (1998),' Automatic text detection and tracking in digital video', Maryland University LAMP Technical Report 028, College Park, pp. 1-38.
[11] Lienhart R and Wernicke A (2002), 'Localizing and segmenting text in images and videos', IEEE Transactions on Circuits and Systems for Video Technology 12,USA, pp. 256–268.
[12] Luo B, Tang X, Liu J and Zhang H (2003), 'Video caption detection and extraction using temporal feature vector', International Conference on Image Processing, Spain, pp. 297–300.
[13] Mallat S.G (1989), 'A theory for multiresolution signal decomposition: the wavelet representation', IEEE Transactions on PAMI 11, New York, pp. 674–693.
[14] Sato T, Kanade T, Hughes E.K and Smith M.A (1998), 'Video OCR for digital news archives', IEEE Workshop on Content Based Access of Image and Video Databases, Bombay, pp. 52-60.
[15] Sato T, Kanade T, Jughes E.K, Smith M.A and Satoh S (1999), 'Video OCR: indexing digital news libraries by recognition of superimposed captions', ACM Multimedia Systems: Special Issue on Video Libraries 7, Japan, pp. 385–395.
[16] Smith M.A and Ksanade T (1995), 'Video skimming for quick browsing based on audio and image characterization', Carnegie Mellon University, Pittsburgh, PA, Technical Report CMU-CS-95-186, pp. 1-22.
[17] Sobottka K, Bunke H and Kronenberg H (1999), 'Identification of text on colored book and journal covers', International Conference on Document Analysis and Recognition, Switzerland, pp.57–63.