Finding Text Regions Using Localised Measures

P. Clark and M. Mirmehdi
Department of Computer Science,
University of Bristol,
Bristol, UK, BS8 1UB,
{pclark,majid}@cs.bris.ac.uk

Abstract

We present a method based on statistical properties of local image neighbourhoods for the location of text in real-scene images. This has applications in robot vision, and desktop and wearable computing. The statistical measures we describe extract properties of the image which characterise text, invariant to a large degree to the orientation, scale or colour of the text in the scene. The measures are employed by a neural network to classify regions of an image as text or non-text. We thus avoid the use of different thresholds for the various situations we expect, including when text is too small to read, or when the text plane is not fronto-parallel to the camera. We briefly discuss applications and the possibility of recovery of the text for optical character recognition.

1 Introduction

Automatic location and digitisation of text in arbitrary scenes, where the text may or may not be fronto-parallel to the viewing plane, is an area of computer vision which has not yet been extensively researched. The problems involved are to first locate the text, then align it correctly to obtain a fronto-parallel view, and finally pass it to an OCR system or a human observer for higher level interpretation. In this paper we are concerned with the first stage of this task.

The research into retrieval of text from 3D scenes has applications for navigating robots that need to gain information from the text in their surroundings, replacing the document scanner with a point and click camera, as an aid for the visually impaired, general Wearable Computing tasks benefiting from knowledge of local text, and other automated tasks requiring the ability to read where it is not possible to use a scanner.

A major area of recognition of text in non-fronto-parallel views is number plate recognition. Cui et al. [4] initially locate a licence-plate in an image using the assumptions that the plate is black-on-white, and has high horizontal spatial variance. They then track features of the plate's characters over a sequence of images and use this to correct the plate's perspective distortion. Barroso et al. [1] locate the number plate by examining the troughs and peaks in horizontal cross-sections of the image. They segment the characters using projection profiles. In most examples of this application area much of the activity is based around useful constraints and assumptions of the orientation of the text, its colour and approximate size.

In other related work, Messelodi and Modena [7] extract lines of text of unknown orientation from images of book covers. They initially threshold the image and then apply a heuristic filter to the resulting binary regions to reject those not associated with text. The

BMVC 2000 doi:10.5244/C.14.68

image is then repeatedly split until separate paragraphs are found. The orientation of each paragraph is estimated by finding the projection profile with the minimum entropy, and this is used to separate text lines. Their approach works well but the text being examined was on a fronto-parallel plane to the camera. The projection profile may not perform so well on text under a perspective transformation. Wu et al. [8] use K-means clustering of the average local energy of an image's derivatives to differentiate the text regions from the rest of the image. This is performed for three different sizes of Gaussian derivative filters at different angles to find text of different size and orientation. In their experiments, all image pixels were classified into one of $K = 3$ segments. Their further processing to identify the text strings within this segment is dependent on the assumption that the text is orientated horizontally in the image (in other words, not skewed or rotated). Li et al. [6] use different moments of wavelet data applied at different resolutions and classified using a neural network to locate text such as film credits and overlaid text, and also some fronto-parallel horizontal text embedded in the original scene. Chen and Chen [2] made some interesting observations when they tried to differentiate text regions from graphics on journal covers. They found that across a text region there is a low variance in the "spatial density", i.e. the ratio of text to background pixels. This is due to the even spacing of text, and the fact that most characters have a similar ratio of their area to the amount of space they take up. They noted that the distribution of edge angles in a text region has peaks at 0° and 90°, due to the large number of horizontal and vertical edges that appear in characters.

The research mentioned above are amongst many other examples which assume the text to be face-on in the image, usually a scanned document. Much of the body of work on text identification for document processing is not applicable to images of the real world. In our work we wish to locate text which is at an orientation to the camera, and embedded in a real-world scene. The camera may be hand-held or be positioned on the body in the context of wearable computing. In [3] we presented a method to locate as well as recover the fronto-parallel view of all regions of text in the image by first extracting local information such as page borders and edges around text. While this method provided good results, the edge extraction and line finding stages of the proposed technique relied on thresholds that can vary from one scenario to another. The main assumption in [3] is that each document in the scene has borders to be recovered. However, some documents may be overlaid or their edges may not contrast well enough against their background to provide the required rectangular frame. Here, we therefore report on an alternative method to locate regions of text which eliminates such problems and which is based around the local image statistics. We combine a number of locally performed measures and use a neural network to classify the text regions. As part of future work, we plan to use the characteristics of each text region to first locate its plane's vanishing points, and then project the region on the plane on which the text lies, to create a fronto-parallel view.

In this paper, the focus is on the location of a wide range of text, from single words and lines of text to larger paragraphs and blocks. As well as highly visible text, we also desire recognition of any text which may not be readable due to being too small in the image, or at too extreme an angle to the camera. This could facilitate an autonomous robot to decide to move into a suitable position to read the text, or a computer controlled camera (wearable or otherwise) which can zoom in on the text in order to read it. The advantage this approach gives these applications is that the resolution of the camera may be minimised.

2 Statistical Measures

We wish to locate all regions of text in greylevel images of real-world scenes, under variable lighting conditions. The human visual system can quickly identify text-like regions without having to examine individual characters, even when the text is too far away to read. This is because text has textural properties that differentiate it from most of the rest of a scene. We now present five statistical measures geared towards identifying specific properties of visible text that can differentiate it from most other parts of an image. The measures $M_s, s \in \{1..5\}$ will be applied to each input image and a neural network will use them to determine likely text-regions in the image.

Each of the statistical measures considered here responds differently to different properties of text. The measures M_s are engaged in small neighbourhoods across the image. For each measure a new image is generated where each pixel in the new image represents the result of the measurement applied to the neighbourhood of the corresponding pixel in the original image. The values chosen for the radii of the neighbourhood masks employed vary for each measure and are discussed later. Figure 1(a) will be used as a running example to illustrate the application of each measure.

Measure M_1: The variance of the greylevel histogram H over a circular neighbourhood of radius 3 (total area $N = 29$ pixels) at each pixel is used as a measure of how much local information there is:

$$M_1 = \sum_{i=1}^{N} (H(i) - \bar{H})^2 \tag{1}$$

where $\bar{H}$ is the mean intensity of histogram H. We are interested in areas of medium variance since text has information, but small and medium scale text undergoes aliasing at the boundaries where text and background greylevels mix, which results in regions of not vastly contrasting intensities. High variance regions generally indicate extreme high frequency changes, such as a single sharp edge. A visualisation of the output of this measure is shown in Figure 1(b).

Measure M_2: Text regions have a high density of edges. This density is measured in a circular neighbourhood of radius 6 centred at each pixel by summing all edge magnitudes located with a Sobel filter:

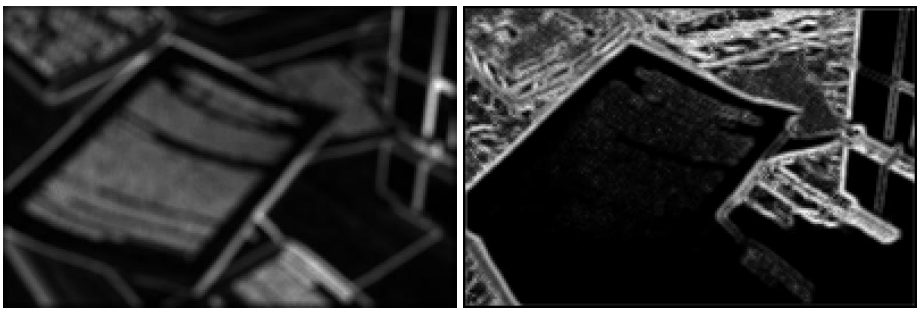
$$M_2 = \sum_{i=1}^{M} E(i) \tag{2}$$

where $E(i)$ is the edge magnitude at pixel i, and $M = 113$ is the number of pixels in the window. Although this measure is similar to the variance of measure M_1, the visualisation shown in Figure 1(c) demonstrates that it is more invariant to changes in lighting (that can be seen in Figure 1(a)).

Measure M_3: Chen and Chen's [2] continuous spatial density assumption (given a flatbed scanner view of a document) states that the ratio of text to non-text intensity greylevels should not vary greatly as we pass over a text region. We apply this principle and hypothesise that that there will be only a small change in local greylevel histograms across a text region (the histograms computed for measure M_1 are reused here). The distance between histogram H and its eight-connected neighbouring histograms G_i is computed as:

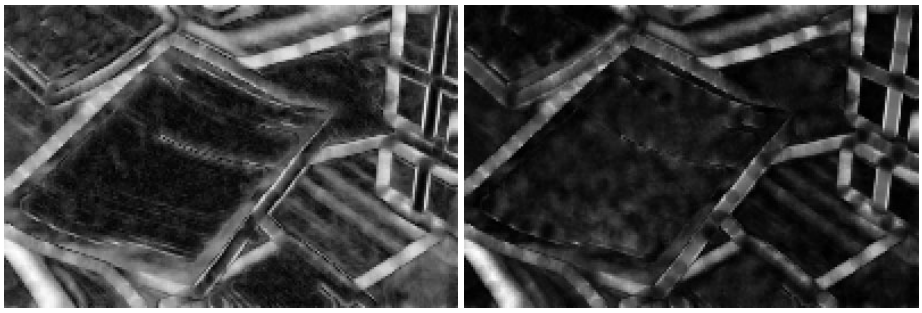
(a) Original image

(b) Measure M_1 output



(c) Measure M_2 output

(d) Measure M_3 output



(e) Measure M_4 output

(f) Measure M_5 output

Figure 1: Example image and visualisation of the results of applying each of the five statistical measures.

$$M_3 = \sum_{i=1}^{8} \sum_{j=1}^{B} (H(j) - G_i(j))^2 \tag{3}$$

where B is the number of histogram bins. By evaluating the difference between one region and its neighbours, the stability of the spatial density is found. The measure produces results like those in Figure 1(d) which shows little change across the text regions.

Measure M_4: In high resolution images one expects to find a high number of edges in a text region, and the angles of the edges to be well distributed due to the presence of curves on many characters. However, this will not be the case at low resolution, where individual characters merge and edges follow the tops and bottoms of text lines. Figure 2 shows the distribution of edge angles in the large text region of Figure 1(a). The angle of an edge is determined by the direction of the gradient of the image at that pixel. We observe that there is a tendency for the magnitude of edges in one direction to be matched by edges in the opposite direction of equal magnitude. More specifically, each edge of a character is likely to be accompanied by an edge in the opposite direction, found on the opposite side of the text character or stroke. We draw the hypothesis that over a text region the histogram of the edge angles has rotational symmetry. Hence, M_4 is a measure of the strength of *asymmetry* using a localised edge angle histogram, A:

$$M_4 = \frac{1}{E} \sum_{\theta=0}^{\pi} (A(\theta) - A(\theta + \pi))^2 \tag{4}$$

where $A(\theta)$ is the total magnitude of edges in direction θ, and E is the overall edge magnitude which normalises the result. θ is incremented in steps of $\frac{\pi}{8}$ which was found to be an adequate resolution. This is performed across the image in a circular neighbourhood of radius 16 centred at each pixel. The image in Figure 1(e) illustrates the results of applying measure M_4. It responds well at strong edges such as page borders and other non-text image structures due to the bias of edges in one direction. Text areas, on the other hand, have a very low strength outcome.

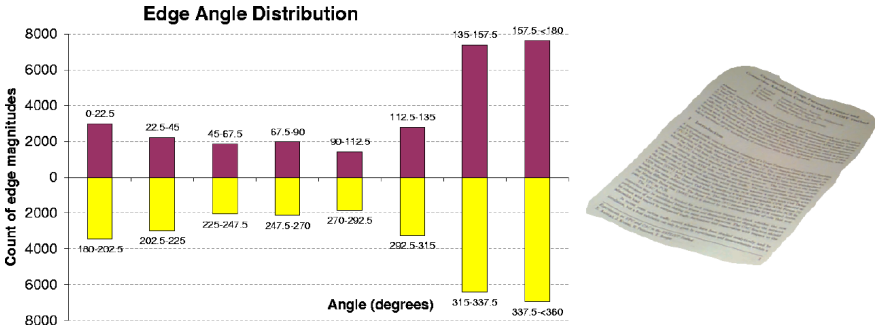


Figure 2: Histogram of edge angle values between $0° - 360°$ for the large text region from Figure 1, shown here on the right.

Measure M_5: As well as cancelling well, we also expect edges in a text region to be well distributed. The first four measures respond in the same way to straight image features as to coarse or curved features. This measure is employed to reject those areas of the image with tight distributions of edges corresponding to straight ramps, canals or ridges in the image. It examines how evenly spread the edge magnitudes are *over all the directions*:

$$M_5 = \sum_{\theta=0}^{2\pi} (A(\theta) - \bar{A})^2 \tag{5}$$

where $\bar{A}$ is the average magnitude of all the directions. If this measure returns a large value, then there is a large variance in the edge magnitudes in different orientations, suggesting a dominant orientation. Since text has a coarse or curved perimeter, large responses suggest that the region under examination is not text. We can see in Figure 1(f) how this measure has responded well to the straight lines in the original image. This allows our proposed method to drop regions that otherwise may be regarded by the other measures as containing text.

3 Characteristics of the Measures

None of the measures $M_s, s \in \{1..5\}$ uniquely identifies a text region. Each one also responds to some non-text areas of the image. The measures are designed to complement each other, so that incorrect decisions by one of them can be corrected by others. In Section 4 it will be shown how the measures are combined to classify text regions.

Circular masks are employed for generating the histograms, finding means and searching for edges. The radii of these masks are important. If they are too small then text regions may be broken up where there are gaps between words and paragraphs. If they are too large, different text regions may overlap, small text regions may be missed, and processing time is wasted. The different measures also operate with different mask sizes. For example, measure M_4 requires a larger area of the image than the other measures because it is sensitive to overlapping one half of a text line. The optimum size of the masks depends on the size of the text we are looking for. Multiresolution methods (performing processing at different scales) such as in [8] offer one solution to this problem. Alternatively, to scan at a higher or lower scale we can change the size of our masks. For the experiments reported here the radii were determined empirically to work for medium sized text in the image and are kept constant for all the images used. However, the training presented in the next section was applied for all scales of text, and our results reflect reasonable recognition across a wide range of text sizes.

4 Combining Measures

The outputs of the five measures can be thresholded and then combined with a boolean AND operation to produce a new image with all the text regions classified. This is not a stable approach due to variation in the measure caused by scene properties such as illumination, and also due to the loss of information caused by considering each measure separately. It is also preferable to avoid the use of thresholds. Instead, we have introduced

a three-layer neural network to use the data from all of the measures simultaneously and make a classification based on the combination of measure values for each pixel. The five measures are provided as inputs to the network, and the final result is a total classification of the image into text and non-text regions.

The measures are normalised before input to the network to have zero mean and standard deviation of 1. This avoids the network having to learn the different distributions of values for each measure. Five nodes are provided in the hidden layer to find consistencies and relationships in the distribution of the measures. The network has two output nodes, which compete to classify a pixel as text or non-text. We trained the network by taking measures from 200 positive (text) and 200 negative (non-text) regions from each of 11 hand-labelled images resulting in 4400 training patterns. The size of text in the training images ranged from large text to text that is too small to read but still recognisable to a human, such as the text in the example image in Figure 1(a). The desired outputs were given as probabilities $\{1.0, 0.0\}$ for a text region, and $\{0.0, 1.0\}$ for a non-text region. Learning was performed using a standard back-propagation algorithm for 300 iterations. During testing, each image is scanned using the circular windows. The measures for each pixel are put into the neural network, and each output node returns a probability value. We subtract the non-text probability from the text probability to get a value ranging between -1.0 for non-text to $+1.0$ for text, visualised in Figure 3(a). This result is then smoothed over the local neighbourhood to gain a local consensus (Figure 3(b)), and thresholded at 0.0 to yield the final text or non-text classification (Figure 3(c)).

(a) Neural network output visualisation

(b) Smoothed probability to take a local consensus

(c) Final located text regions

Figure 3: Output from the neural network and final classification.

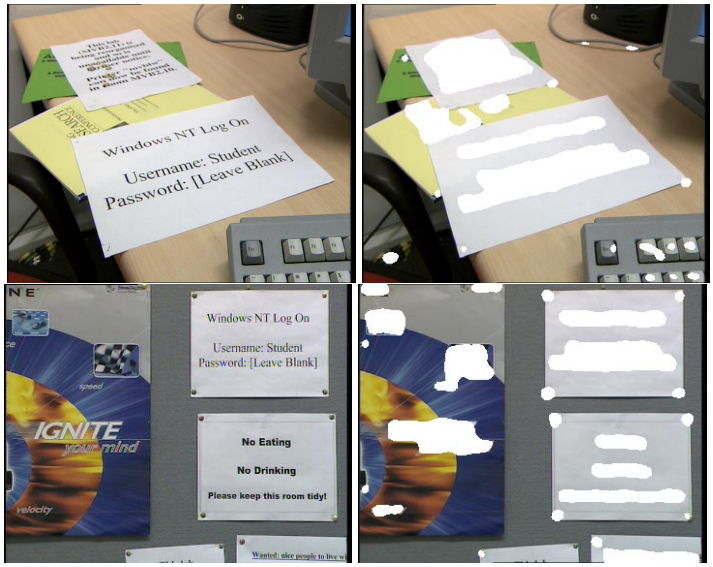


Figure 4: More example images and their located text.

The statistical measures point out areas of the image which are likely to contain text (regions with significant variance), whilst rejecting other areas which cannot contain text due to properties such as an asymmetrical distribution of edges, or over-tightly distributed edge angles. In Figure 3(c) parts of the keyboard region have been picked up as text. This is because the texture of the keyboard has caused the five measures to respond in a similar way to the texture of text. Since the network attempts to deal with all sizes of text simultaneously, it may be confusing the keyboard region with an area of very poor resolution text. In general, the network will often fail to exclude some regions which are not text. While this is an effect we would like to minimise, we prefer to suffer some false positives which can be rejected at a higher level processing stage, rather than miss out any true text regions. Unfortunately, some of the text on the page under the main sheet of paper has not been picked up. It would however be detected in the following frames of the sequence either as the viewer moves into a better viewing angle or gets closer for the size of the text to increase slightly.

Some more classified images are shown in Figures 4 and 5 with the latter containing selected frames from a sequence. It can be seen from the first image in the sequence in Figure 5 that the classification is less accurate at a distance. Although measure M_5 recognises some straight lines on the filing cabinet, the neural network's training on low resolution text suggests that that region of the image could be a very small text line. As the camera moves closer however, the obvious sharpness of the cabinet's edge becomes more apparent, because there are more edge pixels to support that hypothesis. This causes the response of measure M_5 to increase, until it outweighs that of the others, and the network classifies the region as non-text.

For the results shown here, we have thresholded the neural network's output to produce a true/false classification. However, it may be preferable to consider the output as a probability that a region contains text. This corresponds with our own judgement when

perceiving text at a distance. With some of the low-resolution text in our images, even a human cannot tell for sure whether a region of the image actually contains text or just something which appears to be text-like from a distance. In some applications, we may wish to take into account the probabilities as an indicator of which regions are most likely to contain text. This would allow further processing of the image to start with the most favourable regions.

The correct identification of text of an unknown size, including text which is unreadable, is difficult for any algorithm to verify. In fact, we can only ever hope to retrieve a probability that a distant surface contains text. However, at recognisable scales, it may be possible to estimate the size of text from simple image features (under the assumption that text is present) and use this information to guide the network. If text is present, and the estimated scale correct, the measures should be able to make an accurate conclusion. If there is no text, the estimated scale will be irrelevant, and it will be unlikely for the measures to support the hypothesis for text of that size.

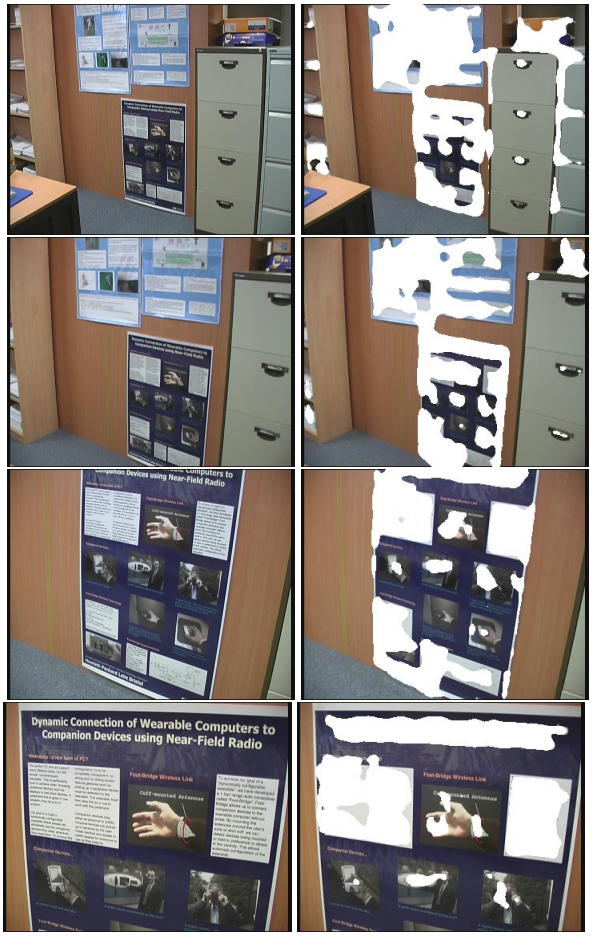


Figure 5: Some results from a sequence of images with the viewer approaching a poster.

5 Conclusions and Future Work

We presented a novel method of finding text regions in images where the document is not aligned on a plane fronto-parallel to the camera view, and the size and greylevel of the text is unknown. Five complementary local pixel neighbourhood measures were introduced. These were fed as input features into a neural network to classify pixels as text. By focusing attention on text regions we can direct higher level processing steps more efficiently. We have avoided the use of thresholds and the parameters we employ, such as circular masks radii, are kept constant throughout. From the results in Figures 4 and 5 it can be seen that small, medium, and large text can be detected in the image. In the future we would like to provide a more detailed analysis of the performance of the technique.

In order to digitise the located text, we need to remove the perspective effects of the text plane and recover a fronto-parallel view of it in readiness for an OCR system. Our perceived method would initially segment paragraphs and lines of text in each local text region, as in [5] or [7]. Once a paragraph has been robustly segmented, its horizontal vanishing point can be calculated as the intersection of the separate lines in the paragraph, and its vertical vanishing points can be calculated either from the paragraph's margins or from the spacing between adjacent lines. With estimates for the vanishing points, we can then recover a face-on view of the paragraph which would be suitable for digitisation by OCR. We are also investigating the use of an active camera which can zoom in on interesting regions for more detailed analysis.

Acknowledgements
The authors would like to thank HP Research Labs, Bristol, UK for their support.

References

[1] J. Barroso, A. Rafael, E. L. Dagless, and J. Bulas-Cruz. Number plate reading using computer vision. *IEEE International Symposium on Industrial Electronics*, 1997.

[2] Wei-Yuan Chen and Shu-Yuan Chen. Adaptive page segmentation for color technical journals cover images. *Image and Vision Computing*, 16(12):855–877, August 1998.

[3] Paul Clark and Majid Mirmehdi. Location and recovery of text on oriented surfaces. *Proc. of SPIE Conference on Document Recognition and Retrieval VII*, pages 267–277, Jan 2000.

[4] Y.T. Cui and Q. Huang. Extracting characters of license plates from video sequences. *MVA*, 10(5-6):308–320, April 1998.

[5] Koichi Kise, Akinori Sato, and Motoi Iwata. Segmentation of page images using the area voronoi diagram. *Document Image Analysis*, 70(3):370–382, June 1998.

[6] Huiping Li, David Doermann, and Omid Kia. Automatic text detection and tracking in digital videos. *IEEE Transactions on Image Processing*, 9(1):147–156, January 2000.

[7] S. Messelodi and C.M. Modena. Automatic identification and skew estimation of text lines in real scene images. *Pattern Recognition*, 32:791–810, November 1999.

[8] Victor Wu, R. Manmatha, and Edward M. Riseman. Finding text in images. In *Proc. 2nd ACM Int. Conf. on Digital Libraries*, pages 3–12. ACM Press, July 23–26 1997.