2011 International Conference on Document Analysis and Recognition

Segmentation and Normalisation in Grapheme Codebooks

Tara Gilliam, Richard C. Wilson, John A. Clark
Department of Computer Science
University of York
York, United Kingdom
{tg, wilson, jac}@cs.york.ac.uk

***Abstract*—The grapheme codebook is a high-performing technique for offline writer identification. This paper considers whether the de facto standards for initial grapheme extraction are optimal for both modern and historical datasets. We examine the construction and representation of the graphemes that comprise the codebook, testing three segmentation methods and two grapheme size normalisation methods on two datasets: a 93-writer IAM dataset, and a 43-writer medieval English dataset. The standard minima-split segmentation is compared to a complementary segmentation method that preserves ligature shapes, as well as the union of both these methods. Classification performance for each method is compared on a range of codebook sizes. We demonstrate that grapheme aspect-ratio is not always a writer-specific feature, and that preserving the character body shape in segmentation is more informative than preserving cursive text ligatures.**

***Index Terms*—Writer identification; Grapheme; Segmentation; Codebook**

I. Offline Writer identification

Given a set of documents written by a group of authors, the writer identification task attempts to label unseen documents with the correct writer. Offline writer identification is a refinement of this task that uses only static images of handwritten text as input and can be used in situations where no movement information is available, such as the analysis of historical documents.

Common feature extraction methods for offline writer identification include slant and edge-hinge distributions [1], [2], [3], run-length distributions [4], [1], Gaussian Mixture Models [5], texture features such as Gabor-based wavelets and filters [6], [7], [8], character-specific structural features [9], [10], and text-fragment shape distributions.

This last method operates by splitting the ink trace into fragments, and selecting a reference set of these to use in generating features for each image. Several variants exist: the most widespread is the grapheme codebook as described in [11], but similar ideas appear in the writer invariants method [12], and at the level of stroke fragments in [13], [14]. Advantages of this approach include high identification performance, text-independence, and automatic adaptation to the script used [2], [15].

The grapheme codebook method is an instance of the bag-of-words strategy for general image matching [16]. A major advantage of this specialisation is a natural and meaningful image segmentation which takes into account the writing structure. The typical segmentation method used assumes a binary input image (black text on a white background), and breaks cursive writing heuristically on the vertical minima of the ink trace. This method was originally given in [17] for Optical Character Recognition and aims to produce the most character-like segments possible, but this occurs at the expense of breaking up the joins between them. Ghiasi and Safabakhsh observe that these joins, or ligatures, between characters contain writer-specific information which can be lost using standard segmentation. They propose an alternate method that combines different sizes of fixed-width segmentation, with good results [15].

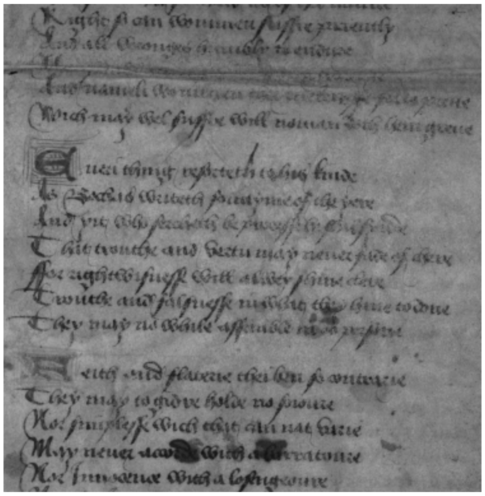


Figure 1. Sample image from the medieval scribes dataset

After segmentation, graphemes can be represented as contours or bitmaps, with little impact on the algorithm performance [18]. Using bitmaps, the grapheme images are usually normalised in size to a uniform 50 x 50 pixels, preserving the aspect ratio. However, Schlapbach and Bunke find that some types of text size normalisation reduce identification accuracy [19]. Fornes et al. find that fixed-ratio normalisation in musical notation consistently gives a higher identification accuracy than normalisation that preserves aspect-ratio [20].

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.129

These results suggest that the typical approach to grapheme extraction may not always be optimal. This work therefore tests alternative methods for the two main aspects of grapheme extraction: segmentation from the cursive ink trace, and size normalisation for grapheme similarity matching. The identification accuracy of the codebook method with these modifications is tested on both a clean benchmark dataset (IAM) and a noisy historical dataset. This allows us to compare whether realistic data responds to these modifications in the same way as a benchmark dataset.

Two grapheme size normalisation methods are tested: square-ratio and aspect-ratio. The square-ratio method scales all graphemes to fill a 50 x 50 pixel square, while aspect-ratio scales by only the largest dimension, preserving the original height:width ratio of the grapheme. We also propose a variable-width segmentation method that complements the minima-split approach by preserving ligatures, and compare them against each other and the combination of both.

The following section describes the datasets used, the general grapheme codebook process, and the particular implementations and methodology used in these experiments.

II. Method and Data

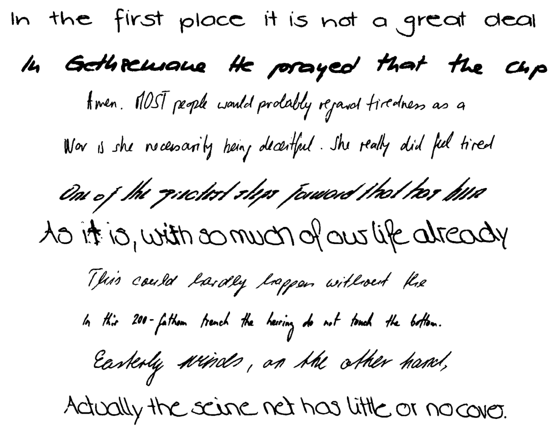
The grapheme codebook method first splits the ink trace of an image into approximately character-level fragments, using some segmentation method. These fragments are called graphemes, and can be stored as either bitmap images or contour representations without loss of performance [18]. A reference set of graphemes is produced by selecting a subset of these – the codebook. Selection can be by Kohonen Self-organising Map [11], k-means clustering [18], or random selection [3]; overall identification accuracy is essentially independent of selection method [3].

The features for each image are formed by measuring the similarity of each of its graphemes to each of the codebook graphemes, and binning it against the closest match. The resulting probability distribution is the sample's feature vector; codebook size determines the dimensionality.

The work in this paper considers the initial process of segmenting and storing the graphemes, and the effect that this has on classification accuracy. In these experiments, codebook sizes of 50, 100, 150, 200, 250, and 500 were chosen for each experiment, as the results in [2] suggest performance peaks at codebooks of around 100 - 400.

Codebook graphemes were selected randomly from the total pool of graphemes generated for a dataset for the given normalisation/segmentation method combination. Grapheme similarity is measured using simple pixel-wise image correlation to generate the feature vectors and the Euclidean-distance nearest-neighbour algorithm is used for classification, which is performed on a leave-one-out basis.

Each experiment was run eight times, with a new set of random codebooks generated for each run, and the mean and standard error of the Top-1 classification accuracy are reported (plotted in Figures 3, 5, 7 and 8).



In the first place it is not a great deal

In Gethsemane He prayed that the cup

Amen. MOST people would probably regard tiredness as a

Nor is she necessarily being deceitful. She really did feel tired

One of the greatest steps forward that has been

As it is, with so much of our life already

This could hardly happen without the

In this 200-fathom trench the herring do not touch the bottom.

Easterly winds, on the other hand,

Actually the seine net has little or no cover.

Figure 2. Sample IAM dataset text lines from [21]

To see how the proposed methods respond to varying sample size and noise levels, all experiments are run on two datasets (total 384 runs). The first is an IAM dataset of the 93 writers made available from the 100-writer identification set [22]. The images are greyscale, containing a single line of text segmented from a copied varied-text paragraph. Image noise is virtually non-existent – standardised recording forms were used, scans are uniform and high-quality, and text lines are cleanly separated, making it an excellent baseline for comparison (Figure 2). The IAM images were processed whole and binarised at a constant threshold using the ImageMagick library[1].

The second is a historical dataset containing approximately 400 full- and part-page images from Middle-English manuscripts, written by 43 scribes. There are between one and 52 images attributed to each scribe; identification of each image was provided by University of York Professor of Medieval English Palaeography, Linne Mooney. The dataset is very irregular, and image noise levels are high. The ink trace is often broken and faded, text lines can be curved or overlapping, and usually both ink and background vary in colour due to ageing or staining (Figure 1). Even where the document is well-preserved, the script within a page can change size, layout and font. The images also vary in size and resolution, from archival quality to samples from a handheld digital camera. In contrast to the IAM set, the medieval dataset is representative of the problems encountered in analysing real-world historical datasets, and required a greater level of processing. Selection and binarisation of the text areas was carried out manually on a per-image basis, with some images requiring additional noise removal. Where necessary, the binarised image was median-filtered to reduce holes in the ink trace. Unfortunately in some cases the original images are low-resolution, leading to some graphemes that are unavoidably jagged.

The remaining sections of this paper describe each experi-

[1] http://www.imagemagick.org/script/index.php

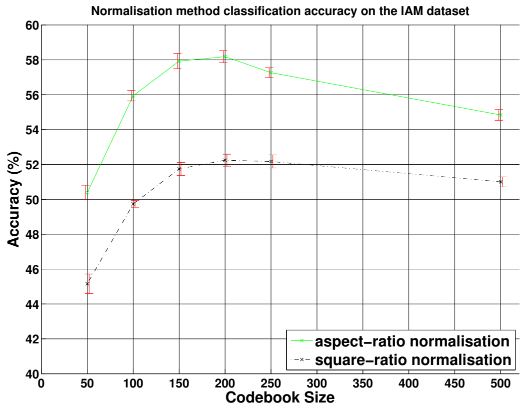


Figure 3. Normalisation results on the IAM dataset

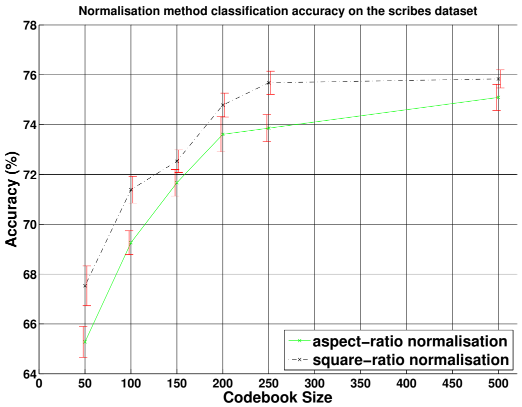


Figure 5. Normalisation results on the medieval dataset

ment and its results in some detail, before concluding.

III. Normalisation

Normalisation methods are specific to the representation of the graphemes, and are essential to allow comparison between writings which vary in size. This experiment considers two possible size normalisation options for grapheme bitmaps: fitting either a single dimension, or both dimensions, to 50 pixels. Figure 4 illustrates the horizontal (columns 1 & 2) and vertical (columns 3 & 4) stretching effect that square normalisation has over the natural aspect ratio of four graphemes from a medieval manuscript image. Preserving the aspect-ratio by scaling in only one dimension retains ratio information that may be writer-characteristic, and (by inspection) appears to be the de facto standard for bitmap normalisation in grapheme codebook experiments [23], [3]. However, some forms of constant scaling in both dimensions has been shown to be beneficial to the writer identification rate, e.g. increasing classification accuracy from 76% to 97% in [19], and providing the best results across all three feature extraction options in [20].

In this experiment, the standard minima segmentation was used to generate two sets of graphemes for each of the scribes and IAM datasets, one aspect-scaled and the other square-scaled. As described in Section II, reference codebooks were generated for each run by randomly drawing from the graphemes generated from the relevant dataset/normalisation combination only. The feature vector generation and classification was identical across experiments in all other respects.

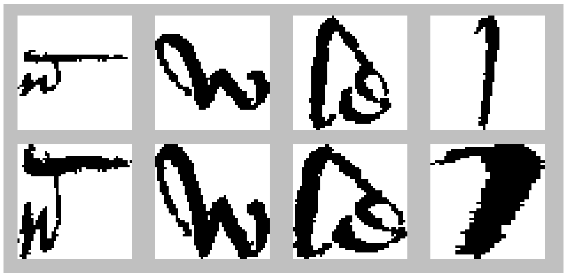


Figure 4. Comparison of graphemes produced by the ratio (top) and square (bottom) grapheme size normalisation methods

Results: Figures 3 and 5 show the variation in Top-1 classification accuracy on each dataset, with error bars of +/- 1 standard error (plotted with some horizontal jitter for clarity). The normalisation experiments on the IAM dataset clearly show that aspect-ratio preservation performs better than fixed-dimension scaling. It produces a highly significant improvement in identification accuracy of 5–6 percentage-points, a substantial effect size equivalent to a boost of 7–11% over the square-scaled accuracy . This effect is fairly constant across all codebook sizes, and confirms that aspect-ratio in freehand Latin scripts carries writer-specific information.

On the medieval scribes dataset, the results are more inconclusive, but suggest that the square normalisation may offer a small (1–2 percentage-point) boost, i.e. aspect-ratio does not carry writer-specific information in this dataset. The reason for this may be that aspect-ratio in character formation is font-specific, rather than directly writer-specific.

Scribes did not typically write in a personal freehand style, but adopted fonts appropriate to the manuscript. These fonts are typical of particular periods and geographic areas, and have a largely fixed aspect-ratio. This implies that aspect is likely to be more strongly correlated with font than with writer in the scribes dataset, as it is limited to scripts produced during the medieval period in England.

The difference in baseline classification accuracy between the datasets is due to the differences in sample size: the scribes dataset images contain up to a page of text (an average of approximately 1000 graphemes), whereas the IAM dataset consists of text-line samples of around 35 graphemes.

Following these results, aspect-ratio normalisation was chosen for the segmentation experiments, as it gives the best overall performance across datasets.

IV. Segmentation

The second experiment compares segmentation heuristics, which determine how the cursive ink trace is split into usable fragments. The standard method to this point aims to approximately divide it into characters, but as we are concerned only with the shape distributions generated by the writer and not the semantic content of the text, this is not a requirement.

In this experiment, the minima method is compared with its complement, which breaks in the centre of characters wherever possible in order to preserve the ligatures instead. Figure 6 shows the difference in splitting points on a connected section of ink trace for each of these methods.

The minima method has been implemented by inserting a vertical break through the minimum inflection points on the lower contour of the ink trace, if it additionally holds that:

- The ink trace height at that point is approximately one stroke-width
- The segmentation will produce a grapheme with a sensible minimum width (set at 5 pixels)

The stroke-width is estimated automatically per-document from the vertical and horizontal run-length distributions.

The assumption implicit in the minima splitting method is that the character body contains the writer-specific information. An alternative hypothesis is that the between-character ligatures contain writer-specific information, and should be preserved.

The implementation of the ligature method initially employs the same minima detection process, but splits instead at the midpoint between adjacent minima (and the connected-component boundaries where necessary). A notable effect of this process is that graphemes are no longer guaranteed to be connected-components themselves.

As these segmentation techniques are complementary, their combination is also tested. To do this, each image in the dataset is represented by the union of the bags of graphemes output by both methods: the raw image data is essentially duplicated, but each copy emphasises a different characteristic. Graphemes identical under both methods are included only once to avoid skewing the feature vector distributions in favour of single characters and small connected-components.

This method distinguishes between the cases where the two splitting methods produce redundant or complementary information: if there is an exact overlap in the information provided, the classification accuracy of the combination should approximately equal whichever single method (minima or ligature) is best. If the two methods are extracting different information, combining should give a classification accuracy greater than either method individually.

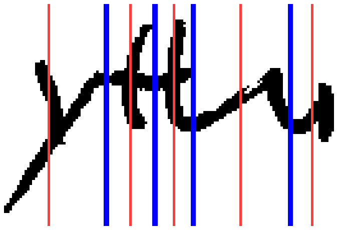


Figure 6. Comparison of splitting points produced by the minima (bold/blue line) and ligature (light/red line) segmentation methods

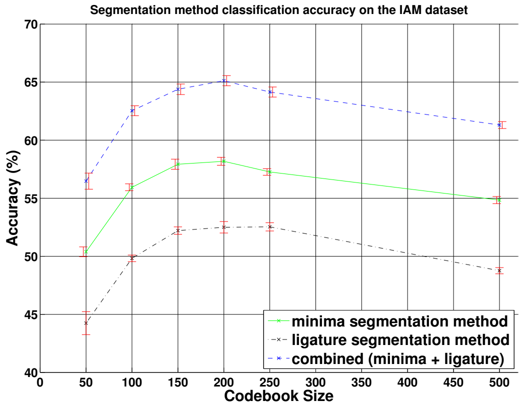


Figure 7. Segmentation results on the IAM dataset

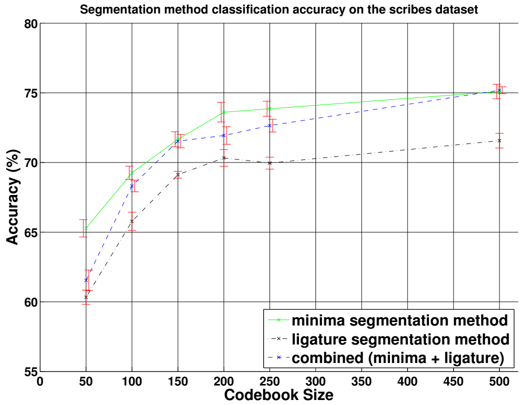


Figure 8. Segmentation results on the medieval dataset

Results: As before, Figures 7 and 8 show the Top-1 classification accuracy on each dataset, with error bars of +/- 1 standard error (plotted with some horizontal jitter for clarity). The segmentation results show that graphemes constructed preserving character ligatures do provide substantial writer-specific information, but the minima segmentation method performs significantly better on both datasets. Again, this result is much less clear on the scribes dataset due to noise effects.

On the IAM dataset, combining the output of both methods gives a significant performance boost, suggesting that the writer information extracted from character body and ligatures is independent to some degree. Identification accuracy for the combined methods increases by 5–6 percentage-points over the minima-segmentation method, and by 12–13 percentage-points over the ligature method. This reflects a substantial proportional accuracy increase of 12% and 25% respectively.

On the scribes dataset, the minima-split method significantly increases accuracy by 3–4 percentage-points, or 5–7% compared to the ligature method. This confirms that the body of the character preserves more writer-specific information than a focus on the between-character ligatures. However in contrast to the IAM dataset, the combined method does not perform significantly differently to the best single-strategy approach. This may be due to the much larger number of graphemes already available per-image, as well as a greater natural variability in results.

V. Conclusions

This work has examined bitmap normalisation and segmentation methods for grapheme codebooks on two very different datasets. Preserving the aspect-ratio of freehand text was found to significantly improve classification accuracy by 7–11% compared to a grapheme size normalisation that discards this information. These results suggest that at the grapheme level, aspect ratio is a writer-specific feature in contemporary freehand writing. However, likely due to the geographic and period influences of font on historical manuscripts, this does not necessarily hold true of historical data: there is at best no increase in performance from aspect-ratio preservation.

In grapheme segmentation for both datasets, preserving solely the character body provides significantly more writer-specific information than preserving solely the between-character ligatures. This effect is greatest on the IAM dataset, with a performance difference of approximately 10%, compared to a difference of approximately 6% for the historical data. Combining multiple splitting methods produces a significant boost in accuracy on the small, clean IAM samples, but the high image noise levels typical of historical datasets may offset any practical gain.

Overall, the standard minima segmentation and aspect-ratio normalisation methods appear to perform well on clean benchmark datasets, but an improvement in identification accuracy can be made for small image samples by combining multiple segmentation methods. However on noisy historical data, the standard aspect-ratio normalisation may have a negative impact, and combining segmentation methods offers no improvement. We conclude that extraction methods appropriate for modern freehand benchmark datasets may not be optimal when applied directly to the increasing numbers of historical datasets in this area.

Future work will include examining the effects of varying sample sizes when combining segmentation methods.

Acknowledgement

The authors would like to thank Professor Linne Mooney of the Department of Medieval Studies, University of York, UK, for providing the data and scribal classifications for this work.

References

[1] M. Bulacu, L. Schomaker, and L. Vuurpijl, "Writer identification using edge-based directional features," in *7th International Conference on Document Analysis and Recognition*, 2003.

[2] M. Bulacu and L. Schomaker, "Text-Independent Writer Identification and Verification Using Textural and Allographic Features," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 701–717, 2007.

[3] L. J. P. van der Maaten, "Improving Automatic Writer Identification," Master's thesis, Maastricht University, 2005.

[4] B. Arazi, "Handwriting identification by means of run-length measurements," *IEEE Trans. Syst., Man and Cybernetics*, vol. 7, no. 12, pp. 878–881, 1977.

[5] A. Schlapbach and H. Bunke, "Off-lineWriter Identification Using Gaussian Mixture Models," in *18th International Conference on Pattern Recognition*, vol. 3, pp. 992–995, 2006.

[6] F. Shahabi and M. Rahmati, "A New Method for Writer Identification of Handwritten Farsi Documents," in *10th International Conference on Document Analysis and Recognition*, pp. 426–430, 2009.

[7] Z. He, B. Fang, J. Du, Y. Y. Tang, and X. You, "A novel method for offline handwriting-based writer identification," in *8th International Conference on Document Analysis and Recognition*, vol. 1, pp. 242–246, 2005.

[8] H. E. S. Said, K. D. Baker, and T. N. Tan, "Personal identification based on handwriting," in *Fourteenth International Conference on Pattern Recognition*, vol. 2, pp. 1761–1764, 1998.

[9] V. Pervouchine and G. Leedham, "Extraction and analysis of forensic document examiner features used for writer identification," *Pattern Recognition*, vol. 40, no. 3, pp. 1004–1013, 2007.

[10] B. Zhang, S. N. Srihari, and S. Lee, "Individuality of handwritten characters," in *7th International Conference on Document Analysis and Recognition*, pp. 1086–1090, 2003.

[11] L. Schomaker and M. Bulacu, "Automatic Writer Identification Using Connected-Component Contours and Edge-Based Features of Upper-Case Western Script," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 787–798, 2004.

[12] A. Nosary, L. Heutte, T. Paquet, and Y. Lecourtier, "Defining writer's invariants to adapt the recognition task," in *Document Analysis and Recognition, 1999. ICDAR '99. Proceedings of the Fifth International Conference on*, pp. 765–768, 1999.

[13] A. Seropian, M. Grimaldi, and N. Vincent, "Writer identification based on the fractal construction of a reference base," in *7th International Conference on Document Analysis and Recognition*, pp. 1163–1167, 2003.

[14] I. Siddiqi and N. Vincent, "Writer Identification in Handwritten Documents," in *9th International Conference on Document Analysis and Recognition*, vol. 1, pp. 108–112, 2007.

[15] G. Ghiasi and R. Safabakhsh, "An Efficient Method for Offline Text Independent Writer Identification," pp. 1245–1248, 2010.

[16] F.-F. Li and P. Perona, "A Bayesian Hierarchical Model for Learning Natural Scene Categories," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 524–531, 2005.

[17] R. G. Casey and E. Lecolinet, "A survey of methods and strategies in character segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 7, pp. 690–706, 1996.

[18] M. Bulacu and L. Schomaker, "A Comparison of Clustering Methods for Writer Identification and Verification," in *8th International Conference on Document Analysis and Recognition*, (Washington, DC, USA), pp. 1275–1279, IEEE Computer Society, 2005.

[19] A. Schlapbach and H. Bunke, "Writer Identification Using an HMM-Based Handwriting Recognition System: To Normalize the Input or Not?," in *Proc. 12th Conf. of the Int. Graphonomics Society*, pp. 138–142, 2005.

[20] A. Fornés, J. Lladós, G. Sánchez, and H. Bunke, "On the Use of Textural Features for Writer Identification in Old Handwritten Music Scores," in *10th International Conference on Document Analysis and Recognition*, pp. 996–1000, 2009.

[21] U. V. Marti and H. Bunke, "Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition systems," *International Journal of Pattern Recognition and Artificial Intelligence*, pp. 65–90, 2002.

[22] A. Schlapbach and H. Bunke, "A writer identification and verification system using HMM based recognizers," *Pattern Analysis and Applications (PAA)*, vol. 10, no. 1, pp. 33–43, 2007.

[23] M. Bulacu and L. Schomaker, "Automatic handwriting identification on medieval documents," in *Proc. of 14th Int. Conf. on Image Analysis and Processing (ICIAP 2007)*, pp. 279–284, 2007.