Iterated Document Content Classification

Chang An, Henry S. Baird and Pingping Xiu

Computer Science & Engineering Dept, Lehigh University
19 Memorial Drive West, Bethlehem, Pennsylvania 18017 USA
Email: cha305@lehigh.edu, baird@cse.lehigh.edu, pix206@lehigh.edu
URL: www.cse.lehigh.edu/~baird

Abstract

We report an improved methodology for training classifiers for document image content extraction, that is, the location and segmentation of regions containing handwriting, machine-printed text, photographs, blank space, etc. Our previous methods classified each individual pixel separately (rather than regions): this avoids the arbitrariness and restrictiveness that result from constraining region shapes (to, e.g., rectangles). However, this policy also allows content classes to vary frequently within small regions, often yielding areas where several content classes are mixed together. This does not reflect the way that real content is organized: typically almost all small local regions are of uniform class. This observation suggested a post-classification methodology which enforces local uniformity without imposing a restricted class of region shapes. We choose features extracted from small local regions (e.g. 4-5 pixels radius) with which we train classifiers that operate on the output of previous classifiers, guided by ground truth. This provides a sequence of post-classifiers, each trained separately on the results of the previous classifier. Experiments on a highly diverse test set of 83 document images show that this method reduces per-pixel classification errors by 23%, and it dramatically increases the occurrence of large contiguous regions of uniform class, thus providing highly usable near-solid 'masks' with which to segment the images into distinct classes. It continues to allow a wide range of complex, non-rectilinear region shapes.

Keywords: *document content extraction, content inventory, layout analysis, shape-oblivious segmentation, uniform content classification, iterated classification*

1 Introduction

We have developed a family of algorithms for document image content extraction, able to find regions containing machine-printed text, handwriting, photographs, etc in images of documents [2, 4, 3, 7]. The vast and rapidly growing scale of document image collections has been compellingly documented[8]. Information extraction[5] and retrieval[6] from document images is an increasingly important R&D field at the interface between document image analysis (DIA) and information retrieval (IR).

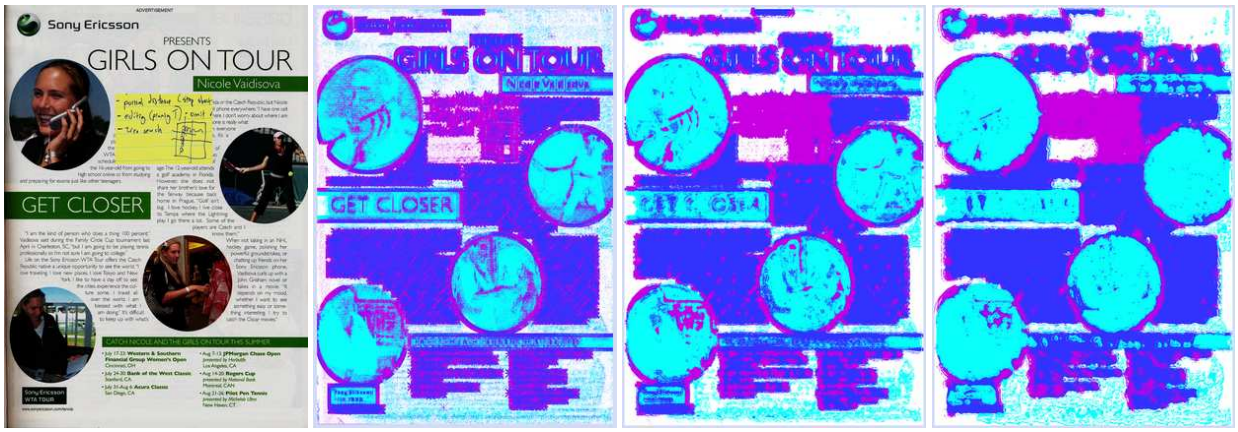
Our content extraction algorithms cope with a rich diversity of document, image, and content types. To date, we have achieved modest per-pixel classification accuracies (of, *e.g.*, 60–70%) which however support usefully high recall and precision rates (of, *e.g.*, 80–90%) for queries on collections of documents[1, 7]. Up until now, we have classified individual *pixels*, not *regions*, in order to avoid the arbitrariness and restrictiveness of limited families of region shapes, as illustrated in Figure 1.

The test image (a) is shown on the upper left (the original image is full color, but is printed in this Proceedings as grey-level). The results of classification are shown to the right (b)-(d), as *classification images* where the content classes are shown in color: machine print (MP) in dark blue (printed as dark grey), handwriting (HW) in red (printed as medium grey), photographs (PH) in light blue-green (printed as light grey), and blank (BL) in white (printed as white). (In this Proceedings, the distinction between MP and HW may be hard to see.) Notice, in the circular regions where PH pixels are located, some MP misclassifications are mixed in: this is an example of a region of non-uniform classification which our method will attempt to correct.

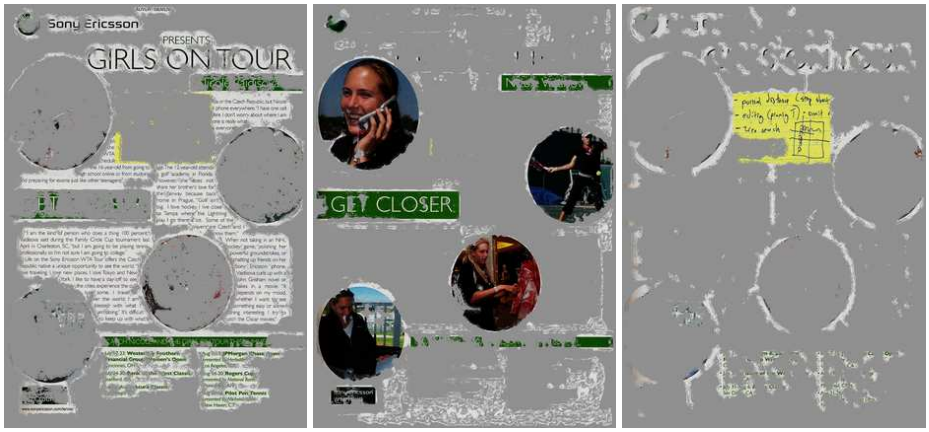
Both training and test datasets consist of pixels labeled with their ground-truth class (one of MP, HW, PH, BL, etc). Each pixel sample is represented by scalar features extracted by image processing of a small region centered on that pixel; these features are discussed in detail in[1]. We have been investigating a wide range of automatically trainable classification technologies, including brute-force 5-Nearest Neighbors (5NN), fast approximate 5NN using hashed k-d trees, classification and regression trees, and locality-sensitive hashing[4, 3, 1]; here, we use approximate 5NN using hashed k-d trees.

2 Experimental Design

In the preliminary experiments reported here, we selected a training set of thirty three images and a distinct test set of eighty



(a) test image (b) 1st stage classification (c) 2nd stage classification (d) 4th stage classification



(e) MP masked (f) PH masked (g) HW masked

Figure 1. A document image with a complex non-rectilinear page layout. Our policy of classifying pixels has the advantage of adapting to arbitrary layouts with non-rectilenear region shapes (here, regions with circular-arc boundaries). The original image (a) is in full color (printed in this Proceedings as grey-level). In the results of classification (b)-(d), machine print (MP) is dark blue (printed as dark grey), handwriting (HW) red (printed as medium grey), photographs (PH) light blue-green (printed as light grey), and blank (BL) white (printed as white). (In this Proceedings, the distinction between MP and HW may be hard to see.) The per-pixel classification error of the 1st-stage classifier is 36.7%; the error of the 2nd-stage classifier is 31.2%; and the error of the 4th-stage classifier is 27.4%. The final MP, PH, and HW masks extract their content types well, as shown in (e)-(g), with the exception a few small patches of HW misclassified as MP.

three images. Together the two sets contain MP, HW, PH, and BL content. Each content type was zoned manually (using closely cropped isothetic rectangles, overlapped where needed to fit non-rectangular regions) and the zones were ground-truthed. The training data was decimated randomly by selecting only one out of every 9000th training sample.

We evaluated performance in two ways: per-pixel accuracy, and subjective segmentation quality.

Per-pixel accuracy: this is the fraction of all pixels in the document image that are correctly classified: that is, whose class label matches the class specified by the ground truth labels of the zones. Unclassified pixels are counted as incorrect. This is an objective and quantitative measure, but it is somewhat arbitrary due to the variety of ways that content can be zoned. Some content—notably handwriting—often cannot easily be described precisely by overlapping rectangular zones. This in some cases will lead to a per-pixel accuracy score being worse than an image may subjectively appear to be.

Subjective segmentation quality. This is a subjective assessment of the quality—expressed as 'good', 'fair', and 'poor'—of the classification as a guide for geometric segmentation of the page area into regions of different content classes. Eventually, automatic methods will be needed to convert a pixel-based classification into one of many possible region-based segmentations: this measure attempts to predict how well that could be done.

3 Design of Post-Classifiers

The goal of post-classification is to enforce local uniformity without imposing arbitrary region shapes. We designed of a trainable post-classifier that operates on the output of the previous classifier, guided by ground truth.

We define the **post-classification** problem as follows:

> *Given*: the per-pixel classification results for a document image and ground-truth.
> *find*: a post-classifier that reassigns classes to favor local uniformity.

Note that the post-classifier also yields a per-pixel classification result for the document image. This inspired us to try *iterated* classification: a sequence of post-classifiers, each trained separately on the training-data results of the previous classifier, guided, as always, by ground truth. We will call the initial stage classifier the *first stage* classifier, the immediately following post-classifier is the called the *second stage* classifier, followed by the *third stage* classifier, etc. Our strategy has been to extract features from small local regions, so that no single classification stage affects a large area. It's worth emphasizing that we train each of the post-classifiers separately on the results from the training set of the previous stage. This strategy appears to prevent the local regions which are dominated by erroneous classes from expanding, while allowing those dominated by correct class to expand slowly.[1]

[1] Before we discovered this, we trained the second stage classifier on the first stage classification results of training set, and used these training samples for all following stages of classification. This allowed local regions that are dominated by one content class to expand, whether the dominant class is correct or incorrect.

For the classification technology, we use approximate 5NN using hashed k-d trees.[3] The features for the post-classifiers are discussed in Section 4.

4 Feature Extraction

Each pixel (the "target pixel") sample is represented by scalar features extracted by image processing of a small region centered on that pixel[1]. After much experimentation we came to rely on these seventy-seven features.

Pixel Class: This feature is the content type value assigned by the earlier-stage classifer to the pixel.

Pixel Content Type: A group of four features, one for each content type (variations on Pixel Class features). E.g., if the classifier labeled the target pixel MP, then this MP feature is set to a non-zero value (186, determined by experiment); otherwise it is set to zero.

Box Class: Four features, one for each content type: the total number of pixels of that content type within a circle of radius 5 centered on the target pixel.

Box Class Euclidean Distance Sum: Four features, one for each content type: each is the sum of all distances from the target pixel to pixels of the content type within a circle of radius 6.

Neighbor Box Class: Sixteen features, four for each content type (variations on Box Class features): each is extracted from the circular regions tangential to the center circle in horizontal and vertical directions.

Box Edge Detection: Thirty-two features, eight for each content type: the total number of pixels of that content type within half of a circle of radius 5 cut in four directions: horizontal, vertical, and the two diagonals.

Encoded Box Edge Detection: Sixteen features (variations on Box Edge Detection features): the difference between two halves of a circle of radius 5.

5 Experimental Results

Our results are illustrated in Figures 1 and 5. Each figure contains seven images of three types: (a) the original image; three classification images from stages one (b), two (c) , and four (d); and three *mask* images for MP (e), PH (f), and HW (f) content classes. In the mask images—say, for example, the MP (machine-print) mask image, only the regions that are classified as machine-print are extracted and displayed using their original color pixel values (printed grey); the pixels of other classes are shown as light grey.

Figure 1 shows results on a color image of a newpaper page containing non-rectilinear handwriting regions. The first stage classifier locates handwriting fairly precisely, but mixes with it many machine-print misclassifications. The post-classifiers significantly enhanced the uniformity of those handwriting regions, after which we could read most of the handwriting extracted by the handwriting mask image. The light blue (printed light grey)

	BL	HW	MP	PH	Type1
BL	0.194	0.061	0.073	0.039	0.173
HW	0.003	0.029	0.006	0.0005	0.009
MP	0.003	0.069	0.202	0.008	0.080
PH	0.002	0.055	0.038	0.208	0.095
Type2	0.008	0.185	0.117	0.047	0.357

Content	*True*	*Classifier*	*Accuracy*
BL	36.7	20.46	94.62
HW	3.842	21.39	13.61
MP	28.37	31.98	63.38
PH	30.71	25.57	81.31

Figure 2. Stage one classifier results on the page shown in Figure 1. The table on the left is the confusion matrix: the rows label ground truth content types; the columns label the content types assigned by the classifier; the 16 entries in the 4×4 top-left subarray sum to 1.0; the Type1 column contains error rates for each true class (that is, the frequency with which that true class is misclassified); the Type2 row gives error rates for each class resulting from classification (that is, the frequency with which that class decision is incorrect); and the bottom right entry gives the overall error rate: 35.7%. Derived from this is the table on the right giving the page inventory—that is, for each *Content* class: the *True* fraction of its pixels classified as that class; the *Classifier*-reported fraction of that class; and the per-pixel *Accuracy* of the classifier on that class. Note that although the per-pixel accuracies on MP and PH are below 85%, the classifier-reported fraction is very close to the true fraction for both of them. In this way, we have often seen, that retrieval based on inventory scores is superior to per-pixel classification accuracy.

	BL	HW	MP	PH	Type1
BL	0.185	0.054	0.076	0.057	0.187
HW	0.0002	0.037	0.0003	0.000	0.0005
MP	0.002	0.030	0.241	0.007	0.038
PH	0.003	0.007	0.031	0.262	0.041
Type2	0.005	0.091	0.107	0.064	0.267

Content	*True*	*Classifier*	*Accuracy*
BL	37.21	19.58	94.73
HW	3.782	12.81	29.08
MP	27.97	34.87	69.24
PH	30.33	32.74	80.02

Figure 3. Stage four post-classifier results on the page shown in Figure 1, including a confusion matrix and inventory as described in Figure 2 above. The per-pixel error rate has fallen from 35.7% to 26.7%.

	BL	HW	MP	PH	Type1
BL	0.171	0.027	0.023	0.005	0.056
HW	0.030	0.024	0.017	0.002	0.050
MP	0.058	0.037	0.343	0.037	0.132
PH	0.022	0.006	0.041	0.152	0.070
Type2	0.110	0.071	0.082	0.045	0.308

	BL	HW	MP	PH	Type1
BL	0.178	0.022	0.022	0.004	0.050
HW	0.015	0.050	0.008	0.001	0.024
MP	0.022	0.035	0.383	0.033	0.091
PH	0.013	0.007	0.034	0.170	0.054
Type2	0.051	0.065	0.064	0.040	0.219

Figure 4. Confusion matrices for stage one and stage four post-classifiers over the entire test set. Post-processing reduces the per-pixel error rate from 30.8% to 21.9%.

texture in the background is uniform from the start and does not worsen under post-classification.

Figure 5 shows results on a color image of a magazine page with a block of handwriting on a yellow ruled background. The iterated post-classifiers cleans much of the sparse light blue texture in the background, without causing the thicker light blue texture to expand, in fact some of it shrinks, which is good. Note that it cleans most of the red texture, both sparse and thick ones, in both the machine print and photo regions. Meanwhile, the curvelinear boundaries of those large regions are accurately detected, as well as the blank regions between paragraphs. The post-classifiers also eliminate most of the erroneous handwriting areas in the yellow ruled background while enhancing the handwriting regions by removing the machine-print texture within them. The mask images are highly promising in representing handwriting, machine-print and photo layers: so we rate the subjective segmentation accuracy as good.

Classification results on the page shown in Figure 1 are summarized in Figure 2 (for the first classifier stage) and 3 (for the fourth stage): the error rate drops from 35.7% to 26.7%. Note the excellent preservation of inventory scores.

The confusion matrices for stage one and four, over the entire test set, is shown in Figure 4. The stage one classifer was, generally, best at recognizing MP and PH, has some difficulty with BL, and has even more trouble with HW, misclassifying 43% of HW pixels as BL.

Figure 6 gives the total error rate in a function of stages of classification. The post-classifiers reduces the error rate by 22.6%. We hypothesize that accuracy will in many cases continue to improve, although perhaps slowly, with more iterations.

6 Instability and Solutions

In one experiment, we ran ten post-classifier stages on a small data set and noticed an instability. For the first eight stages, the total error rate decreased monotonically; then at the ninth stage, errors increased by 26.7% over the eighth stage. Large solid regions of hand-writting were suddenly misclassified as machine-print. The cause appears to be as follows. As post-classification proceeds, local regions become increasingly uniform, whether correctly or incorrectly. It happened that, at the eighth stage, in one training image, a thin "gutter" region separating MP blocks, which was in fact BL, but was, for convenience, manually ground-truthed MP, was classified HW by the eighth classifier. Thus the incorrectly classified samples from these gutters fall at exactly the same point in feature space as correctly classified MP. This led the NN classifer to mistake large areas of MP for HW. The essential problem is that even small incorrectly classified regions, once they are purified above a certain threshold, can compete with large correctly classified regions.

We have found two engineering tricks to reduce the incidence of this instability. The first is to drop a training image out of the training set whenever its classifiaction error rate rises. The second is to increase the radius of the features. We do not yet have a full enough understanding of this problme to propose guaranteed solutions.

7 Discussion and Future Work

We are pleased to see that the total error rate drops by more than 22% even on a large and diverse test set; we expect there is room for further improvement. We plan to experiment with features over a range of scales both smaller and larger than the ones we report here. It is also clear that iterated post-classification frequently enhances the uniformity of regions and yeilds highly useful "masks" for extracting content without imposing an arbitrary and restrictive class of region shapes on the data.

Further improvements may also be achievable by increasing the number of iterations of post-classification.

Acknowledgements

The data base and much of the software architecture is due to Michael Moll. We are grateful for insights and encouragement offered by Michael Moll, Jean Nonnemaker, and Sui-Yu Wang. We acknowledge the continually helpful advice and cooperation of Professor Dan Lopresti, co-director of the Lehigh Pattern Recognition Research laboratory.

References

[1] H. S. Baird, M. A. Moll, and C. An. Document image content inventories. In *Proc., SPIE/IS&T Document Recognition & Retrieval XIV Conf.*, San Jose, CA, January 2007.

[2] H. S. Baird, M. A. Moll, J. Nonnemaker, M. R. Casey, and D. L. Delorenzo. Versatile document image content extraction. In *Proc., SPIE/IS&T Document Recognition & Retrieval XIII Conf.*, San Jose, CA, January 2006.

[3] M. R. Casey. *Fast Approximate Nearest Neighbors*. Computer Science & Engineering Dept, Lehigh University, Bethlehem, Pennsylvania, May 2006. M.S. Thesis; PDF available at `www.cse.lehigh.edu/~baird/students.html`.

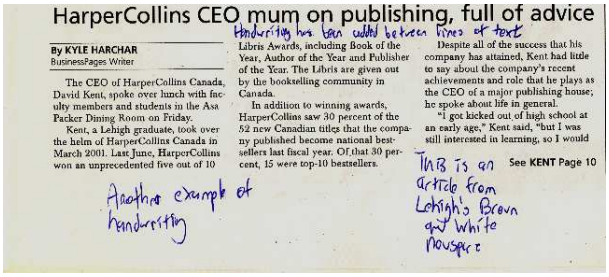
[4] M. R. Casey and H. S. Baird. Towards versatile document analysis systems. In *Proceedings., 7th IAPR Document Analysis Workshop (DAS'06)*, Nelson, New Zealand, February 2006.

[5] Y. Ishitani. Model-based information extraction method tolerant of ocr errors for document images. *icdar*, 00:0908, 2001.

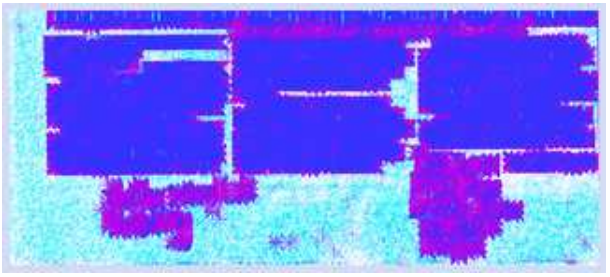
[6] M. Mitra and B. B. Chaudhuri. Information retrieval from documents: A survey. *Information Retrieval*, 2(2–3):141–163, 2000.

[7] M. A. Moll and H. S. Baird. Document content inventory & retrieval. In *Proc., IAPR 9th Int'l Conf. on Document Analysis and Recognition (ICDAR2007)*, Curitiba, Brazil, September 2007.

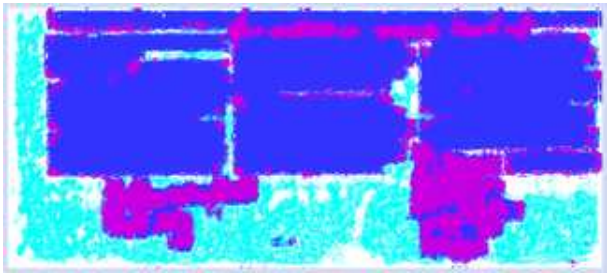
[8] I. H. Witten, A. Moffat, and T. C. Bell. *Managing Gigagytes: Compressing and Indexing Documents and Images*. Morgan Kaufmann, 1999.



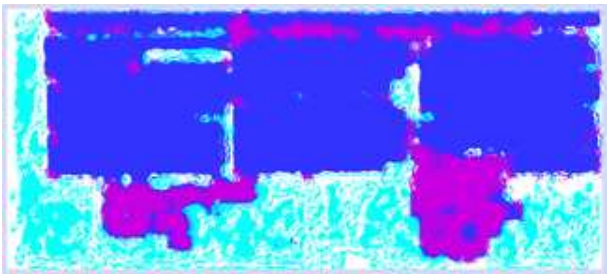
(a) test image



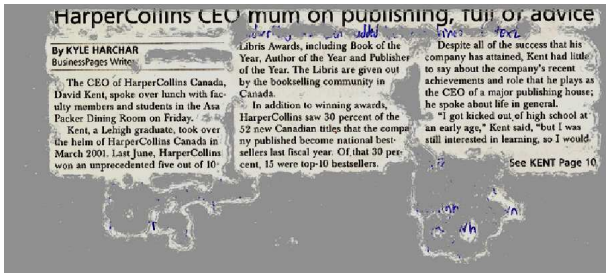
(b) 1st stage classification



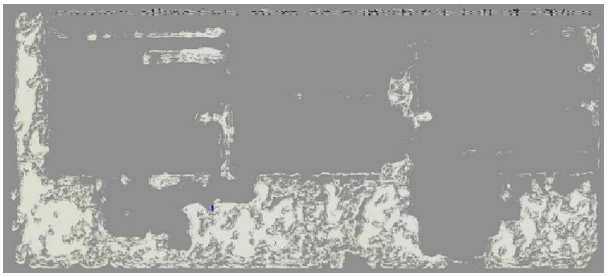
(c) 2nd stage classification



(d) 4th stage classification



(e) MP masked



(f) PH masked

(g) HW masked

Figure 5. A color image containing rectilinear machine-print regions and non-rectilinear hand-writing annotations. The error of the 1st-stage classifier is 37%; the error of the 2nd-stage classifier is 36.4%; and the error of the 4th-stage classifier is 34.2%. The MP mask extracts almost all of the MP except for a little near the (unclassifiable) page boundary. Almost all of the HW is extracted correctly, except for patches where MP crowds it. We rank the subjective segmentation accuracy as fair.

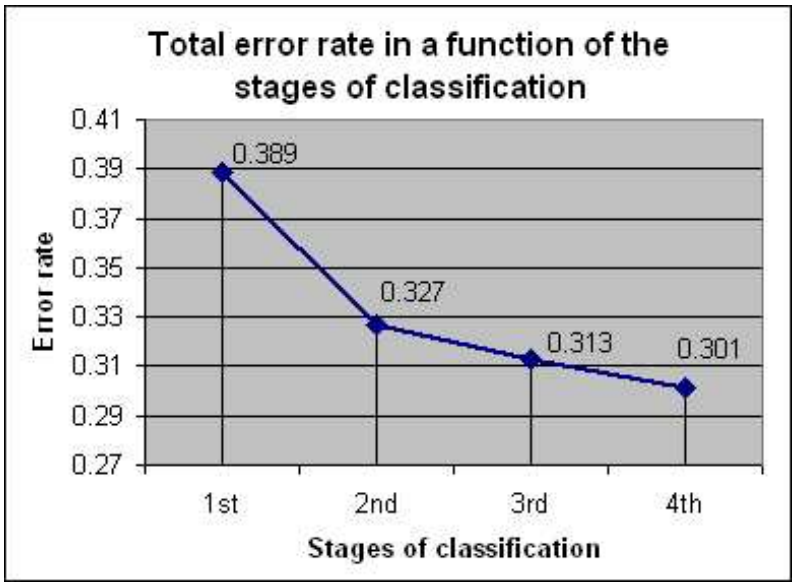


Figure 6. Total error rate averaged over the larger test set, in a function of the stages of classification. After four stages of classification, the error rate has fallen from 0.39 to 0.30, a drop of 23%.