Article Citation Format
Enikuomehin, A.O., Adeseye, I.O, Odugbesan, I., Kanu, O. K. & Sulaiman, A.A. (2020):
Understanding Noise Reduction in Multidimensional documents.
Journal of Digital Innovations & Contemporary Research in Science, Engineering & Technology. Vol. 8, No. 1. Pp 45-56

Article Progress Time Stamps

Article Type: Research Article
Manuscript Received: 9th January, 2020
Review Type: Blind
Final Acceptance: 13th March, 2020

Understanding Noise Reduction in Multidimensional documents

Enikuomehin* A.O., Adeseye* I.O, Odugbesan I., Kanu* O. K. & Sulaiman*** A.A.**
Department of Computer Science,
*Lagos State University, Ojoo, Lagos, Nigeria.
**Adeniran Ogunsanya College of Education, Ijanikin, Lagos, Nigeria
***Ogun State College of Health Technology, Ijese Ijebu, Nigeria

ABSTRACT

The continuous adaptation of computer processes in work places has led to evolving transformation of bound “hard” materials such as book, newspapers etc to soft copies commonly implemented by scanning. However, scanned documents have been very difficult to retrieve largely due to textual noise contained in such documents. Textual noise is known for confusing search models thereby leading to poor retrieval performance. This is more pronounced when considering keyword or cluster based retrieval. When the document is large, suppressing the noise by elimination becomes difficult as the manual processes fail and therefore requires a continuous effective process that can automatically detect and suppress noise in a given retrieval scheme. This research presents an approach for noise minimization over document set using the Bag of Word technique for text terms bagging as an easier way of identifying noisy borders. We applied this technique to some test data of some lecture materials in Lagos State University over an Adhoc retrieval scheme. Our result shows the technique minimized the noise ratio from 85% to 19% and the retrieval efficiency was rated 70%. User’s satisfaction was more of “Appropriate” in all the 67 distinct documents retrieved. The approach will be expanded over a large data set and used as a convention to be built into the output systems of documents scanner.

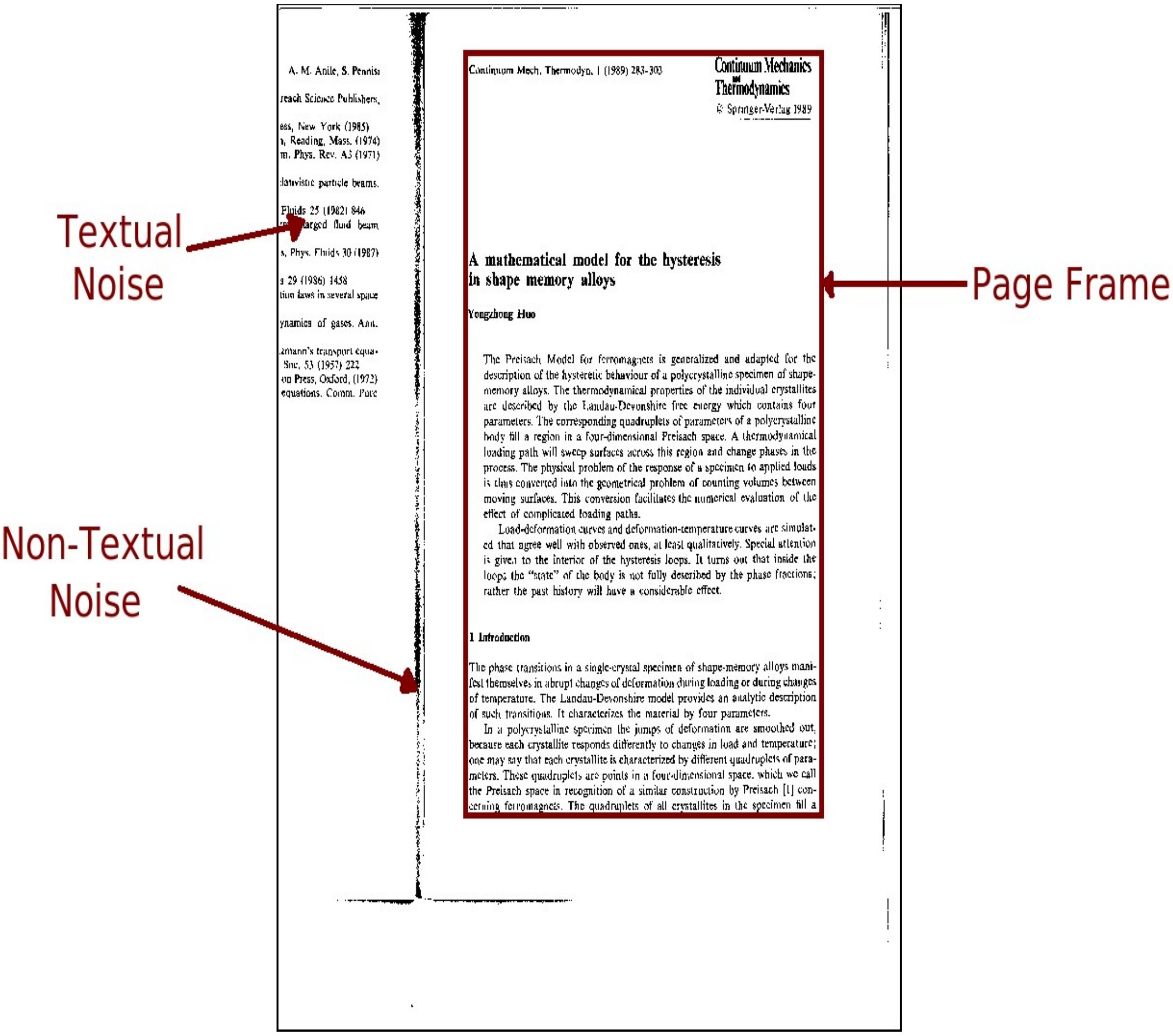
Keywords: Noise, adaptation, tectual noise, noise ratio, multidimensional documents, minimization and scanners.

1. **NTRODUCTION**

The continuous adaptation of computer processes on work places has led to evolving transformation of bound “hard” materials such as books, newspapers etc to soft copies. A common way to do this is by scanning. Scanning a method of changing documents into digital format. However, scanned documents have been very difficult to retrieve largely due to textual noise contained in such documents. Noise can occur in an image (scanned or binarized document) because of the typing machine used, quality of paper used or it can as well be created by scanner when scanning is taking place. Noisy texts electronically saved texts that cannot be classified properly by a software used for text mining. Noisy text is a has differences between the surface form of a coded representation of the text and the intended text. Textual noise is known for confusing search models thereby leading to poor performance. Among other things, noise reduces the accuracy of subsequent tasks of OCR (Optical Character Recognition) systems.(Atena Farahmand, 2013).

It is normally engendered before or after scanning the documents which can appear in the foreground, background or both in some cases.

1.2 Types of Noise in Scanned Documents



A. M. Anile, S. Pennisi
reach Science Publishers,
ess, New York (1985)
ı, Reading, Mass. (1974)
m. Phys. Rev. A3 (1971)
lativistic particle beams.
Fluids 25 (1982) 846
rged fluid beam
s, Phys. Fluids 30 (1987)
s 29 (1986) 1458
tion laws in several space
ynamics of gases. Ann.
mann's transport equa-
Soc, 53 (1957) 222
on Press, Oxford, (1972)
equations. Comm. Pure

Continuum Mech. Thermodyn. 1 (1989) 283-303

Continuum Mechanics and Thermodynamics
© Springer-Verlag 1989

A mathematical model for the hysteresis in shape memory alloys

Yongzhong Huo

The Preisach Model for ferromagnets is generalized and adapted for the description of the hysteretic behaviour of a polycrystalline specimen of shape-memory alloys. The thermodynamical properties of the individual crystallites are described by the Landau-Devonshire free energy which contains four parameters. The corresponding quadruplets of parameters of a polycrystalline body fill a region in a four-dimensional Preisach space. A thermodynamical loading path will sweep surfaces across this region and change phases in the process. The physical problem of the response of a specimen to applied loads is thus converted into the geometrical problem of counting volumes between moving surfaces. This conversion facilitates the numerical evaluation of the effect of complicated loading paths.

Load-deformation curves and deformation-temperature curves are simulated that agree well with observed ones, at least qualitatively. Special attention is given to the interior of the hysteresis loops. It turns out that inside the loops the "state" of the body is not fully described by the phase fractions; rather the past history will have a considerable effect.

1 Introduction

The phase transitions in a single-crystal specimen of shape-memory alloys manifest themselves in abrupt changes of deformation during loading or during changes of temperature. The Landau-Devonshire model provides an analytic description of such transitions. It characterizes the material by four parameters.

In a polycrystalline specimen the jumps of deformation are smoothed out, because each crystallite responds differently to changes in load and temperature; one may say that each crystallite is characterized by different quadruplets of parameters. These quadruplets are points in a four-dimensional space, which we call the Preisach space in recognition of a similar construction by Preisach [1] concerning ferromagnets. The quadruplets of all crystallites in the specimen fill a

Fig. 1: Types of Noise in Scanned Documents

A. Page rule line

Handwritten documents are mostly written on pre-printed, lined papers,.tthe lines can result in the following problems:

v. the ruled lines interfere with the text.
vi. variable thickness in the ruled lines cause problems for the noise removal algorithms.
vii. broken ruled lines cause problems for algorithms detecting them.
viii. Some letters, for example 'Z' , and 'L' which have horizontal lines are removed by the algorithms as they are incapable of detecting differences between them and the horizontal ruled lines. According to Atena et al (Atena Farahmand, 2013), several algorithms have been proposed for ruled line removal.

The methods can be divided into three major categories:

iv. Mathematical morphology -based algorithms which depend on prior knowledge.

v. Algorithms which employ Hough Transform to extract text features and to find lines in every direction

vi. Methods that use projection profiles to estimate lines and hence, reduce the problems' dimensions, which improves the accuracy of the first step in some techniques of noise removal.

Fig. 2: Page rule line

B. Marginal noise

Marginal noises are dark shadows that appear in vertical or horizontal margins of an image. This type of noise is the result of scanning thick documents or the borders of pages in books; it can be textual or non-textual. Algorithms to remove marginal noise can be divided into two categories.

The first category identifies and removes noisy components; the second focuses on identifying the actual content area or page frame of the document.

iii. Identifying Noise Components

The algorithms in this group search for the noise patterns in an image by extracting its features, then remove areas which contain those patterns .Zheng Zhang et al's method employed vertical projection to recover document images that contain marginal noise and decided whether this marginal noise was on the left or right side of the image based on the location peaks in the profile. Then by using extracted features, it detects the boundary between the shadows and cleans the area.

iv. Identifying the Text Components

Another category of algorithms finds the page frame of the document which it defines as the smallest rectangle that encloses all the foreground elements of the document image. This group performs better than the previous one because searching for text pattern is easier than searching for features of noise in a document.

C. Clutter Noise

Clutter noise refers to unwanted foreground content which is typically larger than the text in binary images. This results from numerous sources as punched holes, document image skew or connecting huge amounts of pepper noise.

1.2 Bag-Of-Words Model

The bag-of-words model is model used in representing text data when modeling text with machine learning algorithms such as Neural networks. A problem with modeling text is that it is messy, and techniques like machine learning algorithms prefer well defined fixed-length inputs and outputs(Brownlee, 2017). Machine learning algorithms cannot work with raw text directly; it must be converted into numbers(vectors of numbers). In natural language processing, the vectors are made from textual documents, in order to obtain various linguistic properties of the text. This is referred to as feature encoding or extraction. Bag-of-Words model is a simple and effective technique of feature encoding or extraction. A bag-of-words model extracts features from text for use in modeling using machine learning algorithm. A bag-of-words is a representation of text that describes the occurrence of words within a document. It involves two things:

3) A vocabulary of known words.
4) A measure of the presence of known words.

It is called a "*bag*" of words, because any information about the order or structure of words in the document is discarded. The model is only concerned with whether known words occur in the document, not where in the document(Brownlee, 2017).

1.2.1. Managing Vocabulary

According to Jason(Brownlee, 2017), as the vocabulary size increases, so does the vector representation of textual documents and vice versa. You can imagine that for a very large corpus, such as thousands of books, that the length of the vector might be thousands or millions of positions. Furthermore, each document may contain very few of the known words in the vocabulary. This results in a vector with lots of zero scores, called a sparse vector or sparse representation. Sparse vectors require more memory and computational resources when modeling and the vast number of positions or dimensions can make the modeling process very challenging for traditional algorithms. As such, there is pressure to decrease the size of the vocabulary when using a bag-of-words model.

There are simple text cleaning techniques that can be used as a first step;

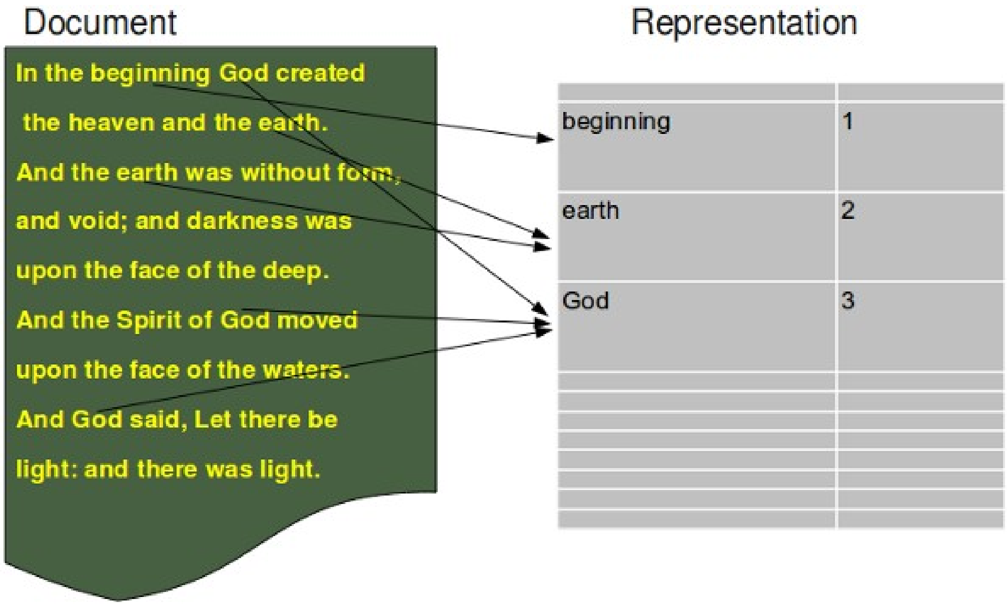
- Ignoring case
- Ignoring punctuation
- Ignoring frequent words that don't contain much information, called stop words, like "a," "of," etc.
- Fixing misspelled words.
- Reducing words to their stem (e.g. "play" from "playing") using stemming algorithms.

A more sophisticated approach is to create a vocabulary of grouped words. This both changes the scope of the vocabulary and allows the bag-of-words to capture a little bit more meaning from the document.

1.2.2 Scoring Words

Once a vocabulary has been built, the occurrence of words in documents needs to be scored. Some simple scoring methods include:

- **Counts**. Count the number of times each word appears in a document.
- **Frequencies**. Calculate the frequency that each word appears in a document out of all the words in the document.



In the beginning God created
the heaven and the earth.
And the earth was without form,
and void; and darkness was
upon the face of the deep.
And the Spirit of God moved
upon the face of the waters.
And God said, Let there be
light: and there was light.

beginning	1
earth	2
God	3

Fig. 3: Document representation using BoW model

1.2.3 Word Hashing

We can use a hash representation of known words in our vocabulary. This addresses the problem of having a very large vocabulary for a large text corpus because we can choose the size of the hash space, which is in turn the size of the vector representation of the document. Words are hashed deterministically to the same integer index in the target hash space. A binary score or count can then be used to score the word. This is called the "*hash trick*" or "*feature hashing*". The challenge is to choose a hash space to accommodate the chosen vocabulary size to minimize the probability of collisions and trade-off sparsity.

1.2.4 TF-IDF

A problem with scoring word frequency is that highly frequent words start to dominate in the document (e.g. larger score), but may not contain as much "informational content" to the model as rarer but perhaps domain specific words. One approach is to rescale the frequency of words by how often they appear in all documents, so that the scores for frequent words like "the" that are also frequent across all documents are penalized.

This approach to scoring is called Term Frequency – Inverse Document Frequency, or TF-IDF for short, where:

- **Term Frequency**: is a scoring of the frequency of the word in the current document.
- **Inverse Document Frequency**: is a scoring of how rare the word is across documents.

The scores are a weighting where not all words are equally as important or interesting.
The scores have the effect of highlighting words that are distinct (contain useful information) in a given document.

1.2.5 Background

As storage becoming cheaper and imaging devices becoming increasingly popular, efforts are made to digitize and archive large quantity of multimedia data (text, audio, image and video). Lots of research is going in the direction for providing as simple as possible data access to end users. In text domain, significant progress happened in that direction. We have search engines like 'Google', 'Bing' etc which provide text search over the web. But on the other hand, search multimedia content is in early research stage.

Most of multimedia contents are retrieved based on tag stored in it. Not only multimedia contents but old manuscripts, books are being digitized. However access to these contents is still a big challenge. Below are the works of researchers in the field of information retrieval/natural language processing.

2. RELATED WORK ON NOISE MINIMIZATION OF BINARY DOCUMENTS

In 2004, Worapoj peerawit and Asanee Kawtrakul (Kawtrakul, 2004), used Edge Density to measure the effect of the edge density based marginal noise removal method, the proposed method was tested on 20 images scanned at 400 dpi taken from the document database from the Kasetsart University Library. These document images were collected from thick books, which were contaminated by variety of marginal noise location and size. The outputs of this method did not only remove marginal noise, but also remove some part of neighbor page. Furthermore, this method can remove some noise by examining all pixels that close to the marginal noise and deleting it. The results of this proposed algorithm shows that all part of marginal noises and some noises that are not in dark regions were removed. If it has the remaining some part of marginal noises, it could be removed by using a general salt-and-pepper noise removal algorithm such as kFill algorithm.

In 2007, F. Shafait et al(Faisal Shafait, 2007),used a geometric matching algorithm to find the optimal page frame, which has the advantages of not assuming the existence of whitespace between noisy borders and actual page contents, and of giving a practical solution to the page frame detection problem without the need for parameter tuning. In geometric matching for page frame detection, a fast labeling algorithm to extract connected components from the document image was used. Two algorithms for extracting text lines and zones from the document were used respectively. After extracting connected components, text lines and zones, the next step was to extract the page frame from document images. Suitable performance measures were defined and the algorithm was evaluated on the UW-III database. The results showed that the error rates were below 4% for each of the performance measures used. Experiments using a commercial OCR system show that the error rate due to elements outside the page frame is reduced from 4.3% to 1.7% on the UW-III dataset. It was shown that the algorithm performs well on all three performance measures with error rates below 4%(i.e 1.7%) in each case. The major source of error was missing isolated page numbers locating the page frame may further decrease the error rates. The benefit of the page frame detection is that the OCR error rates were significantly reduced by removing textual noise.

In 2008, Faisal et al(Faisal Shafait, 2008), proposed an approach for dealing with paper positioning variations in scanned documents. Instead of identifying and removing noisy components themselves, the proposed method focused on identifying the actual content area. This was accomplished by using a geometric matching algorithm. Including page frame detection as a document pre-processing step could help to increase OCR accuracy by removing textual noise from the document. Also in applications like document image retrieval based on layout information, noise regions result in incorrect matches. Using the page frame to reject zones originating from noise can therefore reduce the retrieval error rates. The method for page frame detection took advantage of the structure in a printed document to locate its page frame. This was done in two steps. First, a geometric model was built for the page frame of a scanned document. Then, a geometric matching method was used to find the globally optimal page frame with respect to a defined quality function. The use of geometric matching for page frame detection has several advantages. Instead of devising carefully crafted rules, the page frame detection problem was solved in a more general framework, thus allowing higher performance on a more diverse collection of documents. Additionally, the use of geometric model for page frame detection made the presented approach very robust to the amount of noise present in a document image and could find the page frame even if noise overlaps some regions of the page content area.

The evaluation of the page frame detection algorithm was done on the University of Washington III (UW-III) database. The dataset was divided into 160 training and 1,440 test images. In order to make the results replicable, every tenth image (in alphabetical order) from the dataset was included into the training set. Hence the training set consists of images A00A, A00K, …, W1UA. The training images were used to design the quality function (Sect. 2.3) and to find suitable values for parameters (e.g.). The post processing steps (Sect. 2.5) were also introduced based on results on the training images to cope with different layout styles and the presence of non-textual content at the top or bottom of a page image. The evaluation of our page frame detection system was done on the remaining 1,440 test images. The use of page frame detection successfully detects the page contents region and removes the border noise from the image while keeping the page contents intact.

The researchers presented an algorithm for page frame detection using a geometric matching method. The presented approach did not assume the existence of whitespace between marginal noise and the page frame, and can detect the page frame even if the noise overlaps some regions of the page content area. Several error measures were defined based on area overlap, connected component classification, and ground-truth zone detection accuracy for determining the accuracy of the presented page frame detection algorithm. It was shown that the algorithm performs well on all three performance measures with error rates below 4% in each case. It was also demonstrated that the presented method can handle documents with a very large amount of noise with reasonable accuracy. The error rates on all three performance measures used are below 10% for noise levels up to 80%. The major source of errors was missing isolated page numbers. Locating the page numbers as a separate process and including them in the detected page frame may further decrease the error rates.

In 2014, Faisal Shafait and Thomas M. Brueul (Brueul, 2014), proposed an algorithm for page border noise removal. The algorithm works in three steps:
1) Black filter
2) Connected component removal
3) White filter

Each of these steps are illustrated in the following.

A. Black Filter

The black filter finds large black areas that come as a result of photocopying or scanning and removes them. It looks for these black areas only at the margins of the image so that it does not affect the text or halftones in the center of the image. It uses a rectangular window which moves in these parts of the image, calculating the ratio of black pixels under it at any position and comparing it with a threshold. The rectangular window runs up to 1/3rd of the width or height of the image along the four margins. It starts with the left margin, starting from the x-coordinate = 1/3rd of the image width. The width of the rectangular window is specified by a parameter (default set to 5 pixels). The length or the height of the rectangular window is same as the height of the image. It counts the total number of black pixels under it at any position divided by the total number of pixels under it (equal to width of the rectangular window multiplied by its height) which gives it the ratio of black pixels.

If this ratio is greater than the threshold (default set to 0.70) then it removes everything to the left of itself including itself, and also goes directly on to scanning the next margin(right margin in this case). Else, it moves leftward by the parameter called x-step (default set to 5 pixels) and continues in the same way until it reaches the left border. The rectangular window runs similarly on the right edge starting from 2/3rd of the x coordinate and running up to the right border. It then scans the bottom and the top borders, but while scanning the top and the bottom borders the length of the rectangular window is total width of the image minus the points where it met the threshold while scanning the left and right margins.

For example, if the width of the image is 3300 pixels and while scanning the left border starting at x = 900 it met the threshold and removed (painted white) the entire left margin from x = 0 to x = 900, and while scanning the right border it did not meet the threshold anywhere, so while scanning the top and bottom edges it would scan between x = 900 and x = 3300. The length of the rectangular window for scanning top and bottom edges is chosen like this as the noise beyond the length of the bar has already been considered. When the black pixels ratio goes beyond the threshold while scanning top or bottom edges, it removes the part above or below along the entire width of the image.

B. Connected Component Removal

Connected component analysis first extracts all connected components from the image after applying black filter on it. All components that are very close to the border of the image are considered noise and hence removed from the image. The default border margin set for this purpose is 25 pixels. Hence all connected components whose bounding boxes either start or end within 25 pixels of page border are removed from the image. For scanned documents, this small threshold does not affects components within the page contents area due to the white margin that is typically always present along the border of actual page contents.

C. White Filter

The white filter is very similar to the black filter, the difference being that it removes everything up to the border if it finds a big white block. White filter is run on the image returned after running black filter on the original image and doing a connected component filtering. It uses a different threshold and it runs on slightly different areas of the image. Just like the black filter, white filter also runs on all four margins of the page, but for the left and right margins it starts from x coordinate equal to 1/5th and 4/5th of the image width compared to 1/3rd and 2/3rd for the black filter. For the top margin it starts from 24/25th of the image height and for the bottom margin from 1/50th of the image height. These thresholds are chosen very small in order to prevent the page-footers from being removed as they can be very close to the bottom border. The threshold used for the white filter is 0.995, so that if the number of white pixels are more than 99.5% of the total pixels under the rectangular window, only then the portion is wiped out. The evaluation of their border noise removal algorithm was done on the publicly available University of Washington III (UW-III) database . The database consists of 1600 English document images and is widely used in the document analysis community. The document images in the dataset contain a lot of noise, making it quite suitable for their experiments.

The dataset comes with manually edited ground-truth of bounding boxes for page frame, text and non-text zones, text-lines and words. While the bounding boxes of zones, text-lines, and words tightly enclose their contents, this is not the same for the ground-truth page frame bounding box provided with the data. Instead, there is a margin between the page contents and the ground-truth page frame. In order to prepare ground-truth images for document cleanup task, the ground-truth image might still contain a small portion of border noise if the provided ground-truth is used for cleanup. Therefore they generated the ground-truth documents by using ground-truth zone information. All foreground pixels in the documents that were not contained in any of the ground truth zones, were removed from the image. An example image demonstrating an original UW3 document, its cleanup version using the provided ground-truth page frame, and the cleanup version using ground-truth zones was shown.

The results show that proposed algorithm performs better than the un paper method as pointed out by the Hamming distance metric, but has a slightly lower accuracy than the page frame detection technique . However, the proposed algorithm does not require any pre-processing of the document or extraction of text-lines and zones which makes the proposed algorithm easy to understand and implement. Noise ratio measure shows that unpaper utility removes a larger amount of noise as compared to the page frame detection method or the proposed method. However, this removal comes at an expense of erroneous removal of actual page contents. The large percentage of actual page contents removal by unpaper might make it unsuitable for many practical document analysis applications.

The proposed method, on the other hand, retains more than 99% of the actual page content while reducing the noise ratio from 70% to 20%. It should also be noted that both un paper and page frame detection algorithms were run with their default settings. In 2004, Bruno T. A and Rafael(Lins, 2004) proposed an invading border algorithm which was unfolding in two cases; in the first case, the algorithm assumes that the black noisy border does not invade the black areas of the document. The algorithm was later modified to address the more complex case in which the noisy border reaches the content of the documents. The algorithm was tested over 20,000 images of documents. In at least 95% of them, all border noise was removed, keeping their contents integral. But some of the noisy border still remained in the filtered image.

In 2007, Mudit A. and David(Doermann, 2007) used a novel approach to stroke-like pattern noise(SPN) detection and removal for binary document images. The two-phased approach first understood the script-independent prominent text component features using supervised classification approach. SVM with RBF kernel was used to classify these components from the rest using a minimal set of training samples. Later, based on the cohesiveness and stroke width features of these components, smaller text components are filtered out using K-means clustering. The novelty of the proposed approach was that it does not aim at script or character recognition in order to perform text extraction at diacritic level. It also does not depend on a sufficient number of representative ground-truth samples at component level training. Instead, it uses generic script features to divide-and-conquer components into prominent and dependent ones to achieve noise removal. The approach was limited to Arabic text/scripts not extended to other scripts and documents with mixed content.

In 2019, Umesh et al (Shirdhonar, 2019) used an efficient algorithm for removal of clutter noise from document image. The important operations required were color image to gray scale conversion, image enhancement, gray to binary conversion and clutter noise removal for further processing. The proposed algorithm used document images of publicity available database Tobacco 800 for testing purpose. The document images affected by clutter noise were considered for testing and also some scanned document images the researcher’s database were used to test the proposed algorithm. It was observed that the document was from clutter noise and also the document’s quality was enhanced as an enhanced document with no noise helps in achieving better results during document image analysis. Thus, the proposed framework provides a simple approach for preprosessing the document images.

Omar Boudraa et al (Omar Boudraa, 2019) also proposed algorithm that is based on three processing steps, namely, Preprocessing, Hybrid binarization and Post-processing. In preprocessing, weak contrast is enhanced using CLAHE (Contrast Limited Adaptive Histogram Equalization) method. The proposed model is a novel robust approach for image binarization of degraded historical documents. The algorithm is based on hybrid thresholding using three famous binarization methods, combined with preprocessing and post-processing steps to improve binarization quality. The experimental results prove the effectiveness and the robustness of this method, and show that it achieves high accuracy in document image binarization on three common datasets containing various types of documents which suffer from different kinds of problems and defies (background variation, noise presence, low contrast, etc).

Nevertheless, the method had a major inconvenience, namely: the number of algorithm parameters, which is relatively big (eleven). All of them are set apart by long and separate tests. As a perspective, used Genetic Fuzzy Trees method as proposed by Ernest et al. to control the triggering of sub-algorithms, or the values of our software parameters (i.e.generate Fuzzy Rules). Resorting to other methods of Deep Learning represents another interest idea.

2.1 Related work on Vector Space Model

In 2005, Dr. Khalaf Khatatneh and et.al (ALRIFAI, 2005 - 2010) came up with a new technique to reduce the space used by the information retrieval system using table memorized semiring structure. In this structure, the table includes two arrays one is filled with a word, and the other with some coefficients. This structure implies one document per table, so one row stores the index term and the second row filled with the weight of each index term. Using this new technique will reduce the gab that occur by using the standard matrix in presenting terms and documents ; The number of rows in the matrix will be added by one when adding new document, and the number of columns will be incremented by one in adding new term as well. But in the table memorized semiring structure, any new document has been added to the text collection means a new separate table will be added to the database, and all new terms will be added to the new table.

Khatatneh and et al implied in their research that using the standard matrix will occupy 204248 units to implement vectors, but the new approach will occupy 5388unit which will save more than 198860 space units. In 2009,Michael W. Berry, Zlatko and Elizabeth (Michael W. Berry, 2009) made clear in their research how the fundamental mathematical concepts of linear algebra could be used to manage and index such large documents. they assumed that the traditional indexing techniques is useless; since they takes in consideration when indexing any document all the information inside regardless the actual valuable mean for such extracted terms. For example, the abstracts, author lists, titles key word list and so on are some auxiliary information that are not primarily to understand the content of the paper, those are interested in literature search could find such items useful. Therefore, exclude these items from the indexing process will reduce the capacity consuming and improve the retrieval efficiency. Michael W. Berry and et al in their research used the vector space model mathematical equations to implement such approach. Removing stop words is one of the main phases in the indexing process.

In 2006, Ibrahim Abu El-Khair (El-Khair, 2006) compared in his research the three stop words lists which are used in Arabic language (General stop list, Corpus-based Stop list and Combined Stop list). Using these stop lists with Lemur toolkit and multiple weighting schemes, Ibrahim explored the effect of using stop words on Arabic language retrieval. He used different weighting schemes (BM25, KL and TFIDF) and Recall-Precision performance measurement in order to compare the effect the alternative stop words. The result of such comparison study shows that using BM25 weighting schema with combined or general stop list was the best performing function for Arabic language.There are many other interesting researches concern about data mining and use vector space models. In 2006, Grigoreta and Gabriela (Moldovan, 2006), in their research explained how vector space models could be used in data mining. They presented a new approach that uses clustering in data mining and proposed two techniques: a k-means based clustering technique and a hierarchical agglomerative based clustering technique, and vector space model used for finding the similarity between two methods in order to determine the best over them.

In 2018, Bilal Ahmad Abu-Salih(Abu-Salih, 2018) focused on building vector space model that is valid over Arabic language. “Working with arabic language to build IR systems is very important; since this will enlarge the arabic contents on the web which is major goal that Arabic researches try to reach. One of the magnificent researches related to this is done by Ibrahiem M.M El Emary and Ja’far Atwan (Atwan, 2005).”(Abu-Salih, 2018) Their novel approach was to build such information retrieval that could handle Arabic documents. This comprehensive study used the cosine similarity in the vector space models to compare two technical methods used for retrieving data; these methods are: the full-ward indexing and the root indexing.The technical way in building their systems done by selecting the number of documents as search data. Then, build the stop word table and finally Build the Inverted table. Their experimental results show that examining the system using the full word indexing method by applying 10 queries against 242 of Arabic texts collection make use of the VSM model with cosine similarity measurement, the proposed system retrieves documents in descending order. The output of a query search in full word indexing method while examining the system using root indexing method by applying 10 queries against 242 of Arabic texts collection make use of the VSM model with cosine similarity measurement, the proposed system retrieves documents in descending order. To compare between the two methods in term of which is closer to the user need they used precision as a measurement factor to compare the results in both methods and this showed that using root indexing method will give better results than using full word indexing method. As a conclusion Ibrahiem El Emary and Ja'far Atwan(Atwan, 2005)proved that using root indexing is much useful in IR system because of the following reasons: it decreases the size of storage space, minimize the time needed by the system for processing the documents and query, and gives much amount of retrieved data which may satisfy the user query in best manner.

Dataset

Our proposed algorithm aims at enlarging sample capacity, especially collecting more kinds of sample in order to perfect the system and adopting the system in other domain such as schools (lecture materials in Lagos State University). We further used scribd uploaded lecture note database available at scribd.com, each containing more than 1200 short sentences.

Algorithm:

5) Key points detection through image division or random sampling etc.
6) Local features extraction of the image and generate the descriptor.
7) Clustering of feature related descriptor (usually via K-means) and generating visual vocabulary, in which each clustering centre is a visual word.
8) Summarizing the frequency of each visual word in a histogram.

Images are presented only by the frequency of visual words, which avoids complicated calculation during matching of image local features and shows obvious superiority in image classification with a large number of classes and requiring a lot of training(Brownlee, 2017).

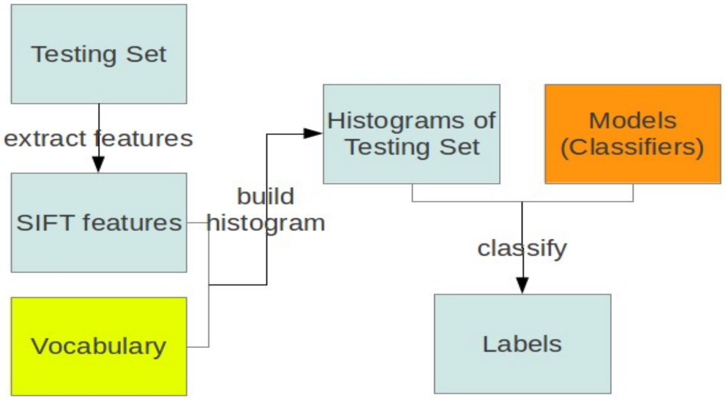


Fig. 4: Block Diagram

4. EXPERIMENTS AND RESULTS

4.1 Data

Two set of data were used for this experiment. First, we considered some photocopied lecture materials of Lagos State University and secondly, Scribd uploaded lecture note database available at scribd.com, each containing more than 1200 short sentences were also used for this experiment.

4.2 Experiment

We applied this technique to some test data of some lecture materials in Lagos State University over an Adhoc retrieval scheme. And the scribd data set on TRECEVAL. The data were manually cleansed before selection was carried out.

4.3 Results

Our result shows the technique minimized the noise ratio relatively from 85% to 19% in and the retrieval efficiency was rated 70%. User's satisfaction was more of "Appropriate" in all the 67 distinct documents retrieved. The approach will be expanded over a large data set and used as a convention to be built into the output systems of documents scanner.

5. CONCLUSSION AND FUTURE WORK

Areas that need researchers to work on to improve the efficiency of this algorithm. We have been able to enlarge sample capacity, especially collecting more kinds of sample in order to perfect(contributed to) the work done by Stipe Celar et al in 2014 and adopted the system in other domain such as school (lecture materials in Lagos State University) and also Scribd uploaded lecture note database available at scribd.com, each containing more than 1200 short sentences. The approach will be expanded over a large set and used as a convention to be built into the output systems of documents scanners.

REFERENCES

18. Abu-Salih, B. A. (2018). Applying Vector Space Model (VSM)Techniques in Information Retrieval for Arabic Language.
19. ALRIFAI, D. K. K. M. W. D. M. A. D. B. (2005 - 2010). Using New Data Structure To Implement documents Vectors In Vector Space Model In Information Retrieval System.
20. Atena Farahmand, A. S. a. J. S. (2013). Document Image noises and Removal Methods.
21. Atwan, I. M. M. e. E. a. J. f. (2005). Designing and Building an Automatic Information Retrieval System for Handling the Arabic Data.
22. Brownlee, J. (2017). A gentle introduction to the Bag-of-Words Model.
23. Brueul, F. S. a. T. M. (2014). A Simple and Effective Approach to Border Noise Removal.
24. Doermann, M. A. a. D. (2007). Stroke-like Pattern Noise Removal in Binary Document Images.
25. El-Khair, I. A. (2006). Effects of Stop words Elimination for Arabic Information Retrieval : A comparative Study.
26. Faisal Shafait, J. V. B., Daniel Keysors and Thomas M. Brueul. (2007). Page Frame Detection for Marginal Noise Removal fron Scanned Documents.
27. Faisal Shafait, J. V. B., Daniel Keysors and Thomas M. Brueul. (2008). Document Cleanup Using Page Frame Detection.
28. Kawtrakul, W. P. a. A. (2004). Marginal Noise Removal from Document using Edge Density.
29. Lins, B. T. A. a. R. D. (2004). A new Algorithm for Removing Noisy Borders from Monochromatic Documents.
30. Michael W. Berry, Z. D., Elizabeth R. Jessup. (2009). Matrices, Vector Spaces and Information Retrieval
31. Moldovan, G. S. (2006). Aspect Mining Using a Vector-Space Model Based Clustering Approach.
32. Omar Boudraa, W. K. H. a. D. M. (2019). Binarization using a Combination of Enhanced Techniques.
33. Shirdhonar, U. D. D. a. M. S. (2019). Preprocessing Framework for Document Image Analysis.
34. Stipe Celar, Z. S., Zljko Marusic and Danijel Zelenika. (2014). Classification of Test Documents Based on Handwritten Student ID's Characteristics.