# *Mitigating the Challenge of Hardcopy Document Forgery*

Nelisiwe Dlamini
Council for Scientific and Industrial Research
Modelling and Digital Science
Pretoria, South Africa
ndlamini2@csir.co.za

Sthembile Mthethwa, Graham Barbour
Council for Scientific and Industrial Research
Modelling and Digital Science
Pretoria, South Africa
smthethwa@csir.co.za, gbarbour@csir.co.za

***Abstract*—The issue of hardcopy document forgery is still prevalent. In South Africa, a series of cases have been reported where academic documents are forged. This poses a problem of integrity, trust and authentication of hardcopy documents. Therefore, it is vital to have a system that could verify a hardcopy document and ensure that the integrity is maintained at all times. Techniques have been introduced to eliminate the issue of document forgery, namely; cryptographic hashing, 2D Barcodes, Digital Signatures and Optical Character Recognition (OCR). In this paper, we present our proposed solution which comprises the 4 techniques. The first part of the implementation included the use of OCR and the experimental results yielded an accuracy of 100%. The experimental setup is described and the overall proposed solution. While this is on-going work, the experimental results demonstrate the feasibility of the proposed solution.**

***Keywords—document integrity; 2D barcodes; optical character recognition (OCR); tesseract; cryptographic hashing; digital signatures; secure hash algorithm (SHA-256); document generation and validation;***

## I. Introduction

An era of digitization has merited the use of modern technology to produce digital documents and preserve these documents. But an inevitable challenge occurs when these documents are printed and issued to different people, since these documents are susceptible to forgery and alterations [1]. A variety of these documents are produced daily, such as academic certificates, wills, case files, birth and marriage certificates, national identity documents, insurance documents, passports and drivers licenses, etc. [2]. A series of forgery cases have been reported in the past. In South Africa, two people presented fraudulent asylum documents at their workplace which they claimed were received from a state’s agency official [3]. Incidents of falsified academic documents have been reported, for instance, in Singapore, where foreign nationals were convicted after being charged with forgery of academic documents [4]. Additionally, in India, a woman who was applying for a passport presented a falsified birth certificate, and it took some time for the concerned agency to detect the forgery. The police reported this as a common occurrence when it comes to applicants [5]. Such cases, indicate the severity of the problem of hardcopy document fraud, and it highlights the importance of securing the original document from being forged in any way.

Fortunately, research in the area of hardcopy document verification is advancing, with the aim of proving that a scanned/physical copy of a document is the same as its original document, that is, that the integrity of the document has not been compromised [1, 2]. Such methods tend to record information pertaining to the original, called the original template. Upon presentation of a copy, recognition techniques attempt to extract this information from the copy, producing a copy template. The two templates are then compared. Two key issues arise. First is the problem of storing the original template, and the second is the problem of extracting the copy template.

While the original template can be stored in a database, an ideal solution would be to store the original template as visible information on the document original itself. Such methods add information to the original document to ensure that a copy is not tampered with in any way. Amongst these methods are watermarks, document signatures, barcodes etc. However, information stored using these methods is limited, for instance an entire document cannot be included in a barcode. Rather than storing the original template on the document, some kind of hash is stored instead [2]. The problem of extracting the stored template from the copy also arises.

The second problem, of extracting the copy template from the copy, is considered in this text. We consider the standard Optical Character Recognition (OCR) approach [6].

For this approach, Tesseract was used, which has been identified as having better accuracy and precision then other OCR techniques, e.g. Transym OCR and GOCR [7]. Despite the praise Tesseract has received, its accuracy is not generally 100%, as our results further confirm, making it difficult to depend on Tesseract as the only solution for verifying a document [7, 8]. Therefore, our aim is to improve Tesseract's accuracy, by (1) limiting the amount of information that requires verification (omitting the details considered as irrelevant), and, (2) rendering such fields in a single “OCR friendly” font, and specifically training Tesseract for this font. In a particular document, information would be classified as

- information that requires verification, and,

- information that should not be verified.

This would limit the process of verification, making it efficient and accurate. If a 100% accuracy is acquired from the use of Tesseract, it means that the process of verifying a copy of an original document would be successful. Achieving 100% accuracy is not only desirable functionally, it also facilitates accurate template hashing.

This paper is organized as follows, in Section 2, we present the Literature Review. A discussion of the experimental work and results, is provided in Section 3. Section 4, discusses the proposed solution we plan to use for the system and Section 5 provides a conclusion to the study.

## II. Literature Review

The challenge of document verification is a highly valued research area, and several techniques have been proposed to mitigate this problem. One of the techniques is the use of watermarking, which aims to preserve the integrity of a document. Watermarking can either be in a digital or printed format [1]. This technique is still vulnerable to attacks, which may not necessarily remove the watermark imprinted, but rather disable its readability [9].

Another method, is the use of Optical Character Recognition (OCR), which is used to recognise characters in a document. OCR is the best tool with regards to character recognition, whereby it takes in an image and returns the recognised text. The main issue with using OCR on its own, is that it is not sufficiently reliable to determine the accuracy (on average the character recognition accuracy is between 90-95%) of a document. Different engines have been introduced that utilize OCR, and for this study we plan to use Tesseract OCR engine, which is considered to be one of the most accurate open source OCR engine [8, 10].

Several research studies have explored the use of two – dimensional (2D) barcodes, whereby information about the document is stored in a barcode and used later for the process of verification [1]. 2D barcodes are commonly used for document verification as they can store more data than 1D barcodes. To strengthen the security of barcodes, various cryptographic techniques are used i.e. public/private key pair, data compression, hash functions, digital signatures [1]. [1] proposed a system whereby, barcodes are used with the help of these cryptographic techniques. Thus, showing the importance of integrating different components to design a suitable solution for the problem of document forgery. A limitation that comes with the use of barcodes is size (the amount of data that could be stored in a barcode). Most of the proposed solutions that utilize barcodes store the entire document in the barcode [11, 12]. To eliminate this issue, our study aims to store only the relevant or important information in the barcodes, which would be identified during the process of document generation.

From all these studies, it can be concluded that efficient techniques must not only be effective but affordable and simple to implement. Thus, this study aims to provide an effective and yet simple and fast method of document integrity verification through the usage 2D barcodes, OCR, digital signatures and cryptographic hashing.

## III. Experimental Work And Results

In this section, a description of the approach and tool used for conducting the experiment is discussed. The tool is divided into two parts, namely; document definition and document validation.

### A. Document Definition

This is a process whereby a document is defined in order to create a dataset. All the documents described are saved as files which is basically a (XML-based) meta-template for the document. The meta-template consists of coordinates which are the height and width of the document to be defined. Each text in the meta-template is labelled with a unique identifier, which makes it easier for the process of validation. The meta-template consists of two types of text; normal text and validation text. The meta-template is used to generate pdf documents and a matching template which only contains validation text as shown in fig. 1. The number of generated documents is specified beforehand. The matching template, is to be used during the process of document validation.

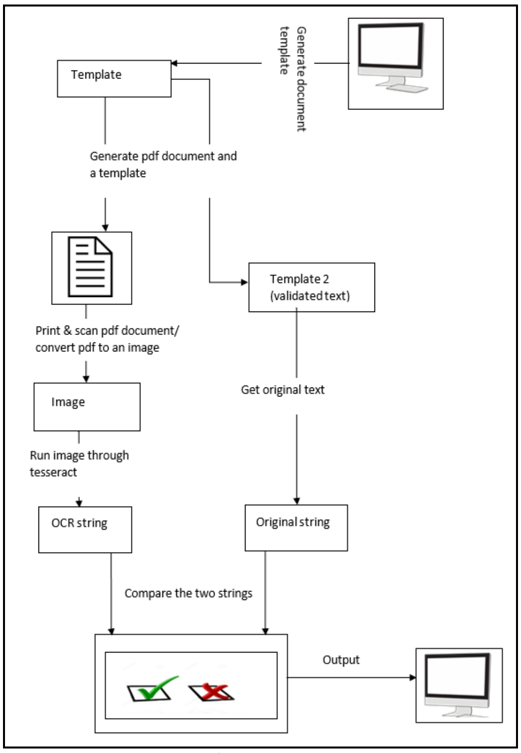
Generate document template
Template
Generate pdf document and a template
Template 2 (validated text)
Print & scan pdf document/ convert pdf to an image
Image
Get original text
Run image through tesseract
OCR string
Original string
Compare the two strings
Output

Fig. 1. Document Definition and Validation Process.

### B. Document Validation

Document validation is the process whereby documents are validated by checking whether a document copy is the same as the original document. Firstly, the tool reads the pdf documents and converts them to images with a resolution of 300dpi in order to maintain the same resolution throughout the experiment. Alternatively, pdf documents are printed, scanned and saved as images with a resolution of 300dpi. To validate, the tool reads in the image and passes it through Tesseract which returns a string. The string is then compared to the expected string from the matching template as depicted in fig. 1.

### C. *Experimental Setup*

To conduct the experiments, a dataset containing 100 different documents was generated. All the generated documents are defined with a border around the edges. The experiments were divided into 3 categories, which are described below.

- Whole document – whereby everything on the document is validated.
- Text labels – whereby each line on the document is validated. All the documents are defined in such a way that the number of text lines are the same. Thus, each document is defined with 35 text lines which is 3500 text lines for 100 documents.
- Characters – whereby each character on the document is validated, which resulted to approximately 40000 characters in total.

Furthermore, the experiments were sub-divided into two approaches:

- Case sensitive - whereby spaces are removed from both expected text and OCR text, without changing the text cases.
- Case insensitive - whereby spaces are removed from both expected text and OCR text; and the case is also changed to upper case. The tool validates each text line and ultimately determines the accuracy of results.

### D. *Results*

The first experiment conducted, was for documents generated using Times New Roman font and the results are depicted in Table 1.

Table 1. OCRB Text Labels Results

| Text Type | Total images | Accuracy (%) | Error Rate (%) |
|---|---|---|---|
| Case sensitive | 3500 | 65.57% | 34.43% |
| Case insensitive | 3500 | 97.17% | 2.83% |

The main aim for the experiment was to get approximately 100% accuracy results, as the images used are converted from the original document. However, as depicted in Table 1, an accuracy of 84% was obtained from validating each character, which was not satisfying as these are original documents. The second experiment was conducted for documents generated using OCRB font (a monospaced font that is optimized for OCR applications). Table 2 and 3 presents the results.

Table 2. Times New Roman Experimental Results

| Category | Total Images | Accuracy | Error |
|---|---|---|---|
| Text Labels | 3500 | 70.74% | 29.26% |
| Characters | 38781 | 84.27% | 15.73% |

Table 3. OCRB Character Results

| Text Type | Total Images | Accuracy (%) | Error Rate (%) |
|---|---|---|---|
| Case sensitive | 38612 | 59.08% | 40.92% |
| Case insensitive | 38612 | 88.32% | 11.68% |

Some of the characters that are frequently misrecognized by Tesseract are shown in table 4.

Table 4. Misrecognized Characters

| Original Character | OCR Text |
|---|---|
| G | 9 |
| R | I” |
| o | 0 |
| I | .I |

To improve the results, AnyOCR font optimized for OCR applications which only consists of capital letters and numbers, was used to generate new documents. AnyOCR font is described as the best OCR font [13], and the results are presented in Table 5.

Table 5. AnyOCR Experimental Results

| Font | Type | Total No. of Images | Accuracy (%) |
|---|---|---|---|
| AnyOCR | Whole document | 100 | 100% |
| | Text labels | 3500 | 100% |
| | Characters | 38 362 | 100% |

From the results presented in Table 5, it can be concluded that, out of the 100 generated documents, Tesseract was able to recognise every character on the document. All the 3 conducted experiments yielded a 100% accuracy result. These documents were printed and scanned, and results are depicted in table 6.

Council for Scientific and Industrial Research (CSIR), Modelling and Digital Science (MDS).

Table 6. AnyOCR Scanned Results

| Category | Accuracy (%) | Error Rate (%) |
|---|---|---|
| Whole Document | 96.0% | 4.0% |
| Text Labels | 100% | 0% |
| Characters | 99.997% | 0.003% |

While an accuracy of 100% was achieved using AnyOCR font (for original and scanned documents), most documents use fonts like Arial, Times New Roman etc. Therefore, we plan to extend our system to work with different fonts. As this is ongoing work, our proposed solution is discussed below.

## IV. Proposed Solution

This section describes the proposed solution for verifying a hardcopy document, which includes 4 components. The first component towards the proposed solution is the use of OCR, which has been discussed in the section above and results were depicted. The remaining 3 components are discussed below. These components have been added to the existing or initial solution to add more security to the system. This proposed solution is expected to eliminate the problem of recreating or tampering with the document. Whereby the text that requires to be verified is hashed and appended to the barcode alongside the digital signature. The digital signature is used to provide a creator's authenticity. The proposed solution is designed in a way that, if an attacker tries to tamper with the document, the system can detect those changes. Below is the description of the proposed components.

### A. Components of the proposed solution

*1) Cryptographic hashing: A cryptographic hash takes a message M and maps it into a fixed length hash value or message digest without the use of keys. Cryptographic hashes are used to determine whether or not the corresponding message has been modified [14]. There are a few widely used hashing techniques and the Secure Hash Algorithm (SHA-256) is mostly used. Husain et al proposed using cryptographic hashes as input to digital signatures in order to verify the authenticity of a printed document [1]. Note that a small change to M results in an incomparable hash, so hash based approaches, while producing small templates, tend to be "all-or-nothing" solutions, producing many false negatives (valid copies identified as forgeries).*

*2) Digital Signatures: Digital signatures are used to prove that a message or hardcopy document in this case is originating from the sender. Essentially, verifying the authenticity of the sender. A digital signature takes in the hash value h, encrypts it using the signer's private key to produce the signature. To verify the digital signature, the receiver must decrypt using the associated public key. The use of digital signatures has been proposed for hardcopy document verification by [15].*

*3) Barcodes: Barcodes are used for storing data and 2D barcodes are mostly used as they can store more data than 1D barcodes. QR Code and Data Matrix are the common types of 2D barcodes [15]. This technique has been proposed for document verification by [1, 2, 11, 12]. When data has been added to the barcodes, these are then added to the document. The 2D Barcode information capacity can differ from v1, 21*21 cells to v40, 177*177 cells by adding 4 to the row and column cells of each version. Its capacity grows until it reaches the maximum allowable capacity number [16].*

### B. Proposed Solution Design

The solution consists of 2 main processes; generation and validation process. The generation process includes all the components discussed here. Below is a detailed description of our processes.

*1) The Generation Process: This process is an extension from the previous process discussed in section 3, whereby the above described components are incorporated. Once the document is created with normal and validation text, each validation text is hashed to produce a hash value that is later used to create a digital signature. The digital signature and metadata are encoded to barcodes that are added along the sides of a document. The metadata consists of; position, length, width and checksum of all validated text, hash and timestamp. Finally, the PDF document is generated and printed. This process is ilustrated in fig. 2 and 3.*

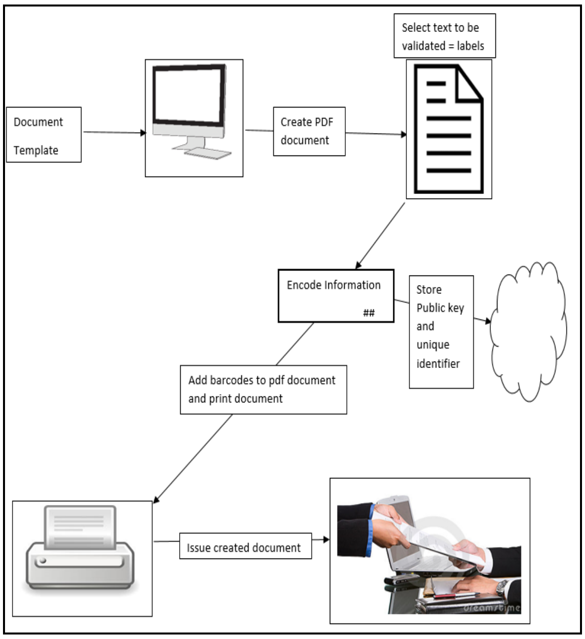
Select text to be validated = labels
Document Template
Create PDF document
Encode Information ##
Store Public key and unique identifier
Add barcodes to pdf document and print document
Issue created document

Fig. 2. Generation Process.

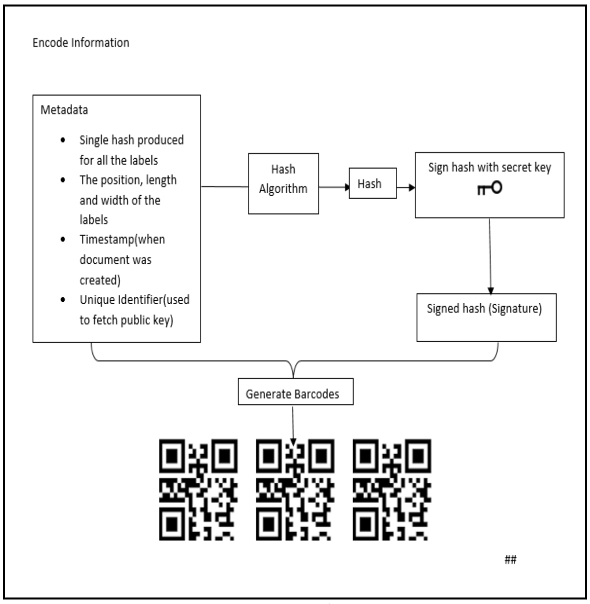
Encode Information
Metadata
Single hash produced for all the labels
The position, length and width of the labels
Timestamp(when document was created)
Unique Identifier(used to fetch public key)
Hash Algorithm
Hash
Sign hash with secret key
Signed hash (Signature)
Generate Barcodes
##

Fig. 3. Encode Information.

*2) The Validation Process: This process is also an extension from the previous process discussed in section 3, whereby barcodes are now accomodated. The documents presented can either be electronic or physical hardcopy format. If a hardcopy format is presented, the document is scanned using a basic scanner. To validate the document, the system reads in the scanned image and the QR codes are identified and decoded. Subsequently, the digital signature is validated, if valid, the meta-data is extracted and used to locate the validated labels. Thereafter, Tesseract is used to validate the text labels. The hash (of each validated label) is calculated and compared with the one from the original document. If the comparison fails, it means the document has been altered. In addition to the hash that is included, a checksum for each text label is also calculated, this aids to point to the exact text label that is not matching. Fig. 4, illustrates the process of document validation.*

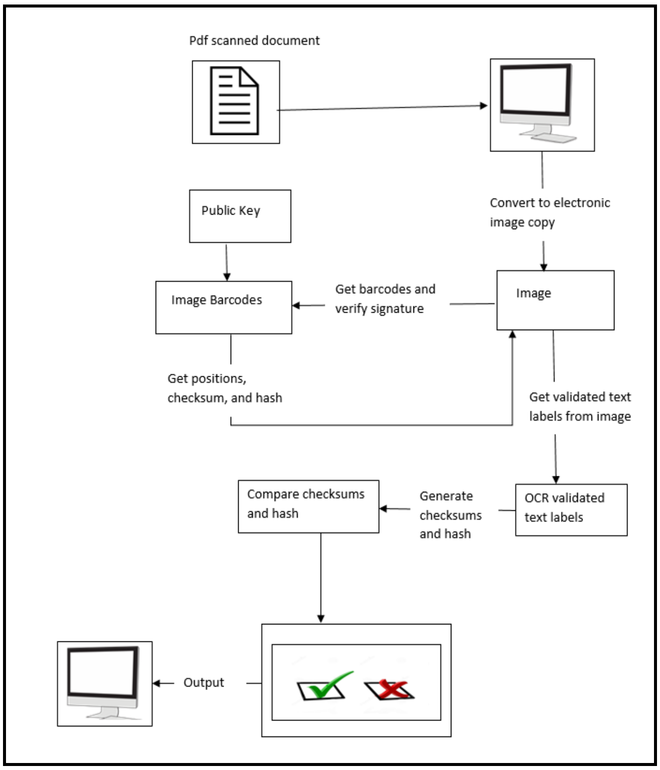
Pdf scanned document
Convert to electronic image copy
Public Key
Image Barcodes
Get barcodes and verify signature
Image
Get positions, checksum, and hash
Get validated text labels from image
Compare checksums and hash
Generate checksums and hash
OCR validated text labels
Output

Fig. 4. Validation Process.

The evaluation of the proposed solution will use a dataset generated using the generation process. The dataset would consist of 100 generated documents, using different fonts, i.e. Courier, Arial, Times New Roman etc., in order to determine the accuracy of the documents. The dataset will also be separated into different groups, on the size of text, which are; small, medium and large text. Usually, most presented documents are damaged. Thus, the dataset would undergo wear and tear. The aim is for our proposed solution to verify the documents regardless of their damaged state.

Based on the studies conducted, the process of generating documents differs, i.e. [11] used Microsoft word to generate documents. Usually the whole document is verified which tends to produce low accuracy results [1, 2, 12]. However, our solution only verifies specified text, hence improving results. Most of the methods discovered through a comprehensive literature study, use similar components, yet they are still proposed solutions with no results that could be used for comparison.

## V. Conclusion

This paper presented our proposed solution for the problem of document forgery, in which 4 techniques are employed; OCR, cryptographic hashing, digital signatures and 2D barcodes. OCR was the first technique to be implemented, whereby documents were generated using a font known as AnyOCR (which is designed for OCR tools) and Tesseract was used to validate the documents. The experimental results yielded an accuracy of 100%, which is good. Since this is still ongoing work, Tesseract is still going to be trained on multiple fonts with the expectation of yielding 100% accuracy. This is to enhance its ability when it comes to handling various fonts. Taking into account that people sometimes present damaged documents, these will also be used in the process of enhancing the robustness and precision of the tool.

In future work, this proposed solution will be implemented and tested for its practicability to detect forgery and ensuring that the integrity and authenticity of a hardcopy document is maintained.

## Acknowledgment

The authors would like to express their gratitude to the CSIR Modelling and Digital Science for sponsoring this research.

## References

[1] A. Husain, M. Bakhtiari, and A. Zainal, "Printed Document Integrity Verification Using Barcode," *Journal Teknologi (Sciences and Eng*, pp.99-106, 2014.

[2] M.H. Eldefrawy, K. Alghathbar, and M.K. Khan, "Hardcopy document authentication based on public key encryption and 2D barcodes," In *Biometrics and Security Technologies (ISBAST), 2012 International Symposium*, pp. 77-81, IEEE, March 2012.

[3] S. Kalipa, "*Home Affairs official sold us fake papers _ IOL News, Crime and Courts IOL News*," Available at: https://www.iol.co.za/news/crime-courts/home-affairs-official-sold-us-fake-papers-1929048 (Accessed: 21 August 2017), 2015.

[4] N. Ganesan, "*Three foreigners jailed in Singapore for submitting fake academic certificates _ Human Resources Online, Human Resources*," Available at: http://www.humanresourcesonline.net/three-foreigners-jailed-for-submitting-fake-academic-certificates/ (Accessed: 16 August 2017), 2017.

[5] A. Rashid, "*Pune Woman submits 'forged' documents to passport authorities twice; probe on _ The Indian Express, The Indian Express*," Available at: http://indianexpress.com/article/india/india-news-india/pune-woman-submits-forged-documents-to-passport-authorities-twice-probe-on-3014055/ (Accessed: 24 August 2017), 2016.

[6] R. Jain, and D. Doermann, "Visualdiff: Document image verification and change detection," In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on* pp. 40-44, IEEE, August 2013.

[7] S. Dhiman, and A. Singh, 2013. "Tesseract vs gocr a comparative study," *International Journal of Recent Technology and Engineering*, *2*(4), pp.80, 2013.

[8] C. Patel, A. Patel, and D. Patel, 2012. Optical character recognition by open source OCR tool tesseract: A case study. *International Journal of Computer Applications, 55*(10), 2012.

[9] S. Oliveira, M. Nascimento, and O. Zaiane, "Digital watermarking: status, limitations and prospects," 2002.

[10] F.Y. Omee, S.S. Himel, M. Bikas, and A. Naser, "A complete workflow for development of bangla OCR," *arXiv preprint arXiv:1204.1198, 2012*.

[11] M. Salleh, and T.C. Yew, "Application of 2D Barcode in Hardcopy Document Verification System," In *ISA*, pp. 644-651, June 2009.

[12] C.M. Li, P. Hu, and W.C. Lau, "Authpaper: Protecting paper-based documents and credentials using authenticated 2D barcodes," In *Communications (ICC), 2015 IEEE International Conference*, pp. 7400-7406, IEEE, June 2015.

[13] M. Jenckel, S.S. Bukhari, and A. Dengel, "AnyOCR: A sequence learning based OCR system for unlabeled historical documents," In *Pattern Recognition (ICPR), 2016 23rd International Conference on* pp. 4035-4040, IEEE, December 2016.

[14] M. Singh, and D. Garg, "Choosing best hashing strategies and hash functions," In *Advance Computing Conference, IACC 2009. IEEE International* pp. 50-55, IEEE, March 2009.

[15] M. Warasart, and P. Kuacharoen, "Paper-based Document Authentication using Digital Signature and QR Code," pp. 94-98, 2012.

[16] N. Victor, "Enhancing the data capacity of qr codes by compressing the data before generation," *International Journal of Computer Applications*, *60*(2), 2012.