Automatic Ascii Art Conversion of Binary Images using NNF and Steganography

Kalpana.C

Abstract--In this application, I have presented a novel application of NMF related methods to the task of automatic ASCII art conversion, where I fit a binary image to a basis constructed from monospace font glyphs using a winner-takes-all assignment.I presented some examples, and demonstrated that when compared to a standard pseudo inverse approach, non-negative constraints minimize the black space of the ASCII art image, producing better defined curves. Furthermore, I propose the use of the divergence cost function for this task, as it provides an element of control over the final ASCII art representation.In the computer world, there is a constant struggle to keep secret information secret, private information private, and when profits are involved, protect the copyrights of data. To accomplish these increasingly difficult tasks, new methods based on the principals of steganography are being developed and used.

Index Terms—ASCII,BMP,LSB,NNF,TEXT File,Encryption,Decryption
.

1 INTRODUCTION

ASCII Generator is a powerful ASCII Art generation application. You can make ASCII Art Words, ASCII Art Photos and even ASCII Art Animations easily by using Convert Image into ASCII Generator.Convert Image into ASCII Generator can take an image and process it to an HTML, RTF, BMP or TEXT file of color-coded text characters, that when combined, resemble an image. It is an ASCII Art Photo. And the files are very worthy of being published to the Web or in the document. Also, you can make your individual ASCII Art Signatures in Convert Image into ASCII Generator. Use them in your e-mails, documents or even in the forums on the web will be a good idea. In Convert Image into ASCII Generator, drawing your own ASCII Art Photos is like drawing a picture in the Paint application of Windows. All these are very easy, no experience need.I have propose a new method for strengthening the security of information through a combination of signal processing, cryptography and steganography

2 HIDING INFORMATION

As much of today's communication is being done over technologically advanced systems (e-mail, instant messaging services, etc.), secrecy of that communication is ever present. The hidden data/file is the message which we wish to keep secret. If data looks random and adding information into this data does not change the randomness, then we have achieved steganography.

Since this byte can contain any value, this implies randomness. By changing the least significant bit (LSB) of any byte within the image file, a human eye viewing the image will not be able to tell a difference from one shade to the next.This allows us to only hide a message one-eighth the size of the original cover file. This is not much if you think that having a cover image of 128 bytes will only yield us a 16 byte hidden message. The growing field of cyber forensics detective work in the digital domain should create greater demand for steganalysis tools in the near future.

3 TO SOLVE THE NON-NEGATIVE MATRIX FACTORIZATION

Non-Negative Matrix Factorization is a method for the decomposition of multivariate data, where a non-negative matrix, V, is approximated as a product of two non-negative matrices, V =WH. NNF is a parts-based approach that makes no statistical assumption about the data. In-stead, it assumes for the domain at hand, e.g. binary images, that negative numbers are physically meaningless—which is the foundation for the assumption that the search for decomposition should be confined to a non-negative space, i.e., non negativity assumption. The lack of statistical assumptions makes it difficult to prove that NNF will give correct decompositions. However, it has been shown in practice to give correct results.

The following procedure for automatic conversion of binary images to ASCII art:

1.Construct W from a monospace font, e.g., Courier, where the glyphs that represent the 95 printable characters (numbered 33 to 126) of the 7-bit ASCII character encoding scheme are stored as M × N bitmaps, which are arranged as vectors of size R and placed in each column, wj . Rescale each column to the unit L2-norm, wj = wj kwjk , j = 1, . . ,R.

2.Partition the binary image X ∈ RP×Q into M × N blocks forming a P/M × Q/N grid, where each block corresponds to a font glyph in the final ASCII art image.Construct V from the blocks by arranging as vectors and placing in columns. If X is not evenly divisible into M×N blocks then perform zero padding to the required dimensions.

- *Kalpana.C is currently pursuing masters degree program in computer science and engineering at SBM College of engineering and technologyDindigul. E-mail: kalpsbabu005@gmail.com*

3.Randomly initialise H; specify β & ...

4.Fit V to W using the H update rule (Eq. 1), and repeat for the desired number of iterations.

$$h_{jk} \leftarrow h_{jk} \frac{\sum_{i=1}^{M} w_{ij}(v_{ik}/[\mathbf{WH}]_{ik}^{2-\beta})}{\sum_{i=1}^{M} w_{ij}[\mathbf{WH}]_{ik}^{\beta-1}}. \quad (1)$$

5. Assign each block location in the original image a glyph based an a winner-takes-all approach, where the maximum value in each column of H corresponds to the winning glyph in W (Eq. 2). Reverse the block partitioning procedure of step 2 and render the ASCII art image using the identified glyphs in the specified monospace font.

$$V \approx W \max col(H, o) \quad (2)$$

A test image (UCD CASL logo) and three ASCII art representations, which are created using the pseudo inverse and NMF utilizing the SED (Squared Euclidean Distance) and KLD (Kullback Leibler Divergence) cost function. Inspection of the logo text reveals that NMF preserves the curves best and minimizes black space. Furthermore, the selection of a different creates a different ASCII art representation

It may be possible to improve the resultant ASCII art representations by finding the most natural grid for the binary image, which may be achieved by shifting the image both vertically and horizontally and fitting the image to W. The grid that results in the best reconstruction, as indicated by the signal-to-noise ratio for example, may be considered to be the most natural grid.

The chosen glyphs in an ASCII art image are selected based on a winner-takes-all approach. It is possible to reduce the number of activations in H by using a sparse NMF algorithm, which may result in less iteration to achieve the same ASCII art representation. For the glyph set used to construct W in our M had the largest amount of black space as indicated by the Frobenius norm. However, M was not chosen as the fully black block glyph using any of the presented cost functions, which suggests that a more suitable cost function exists.

The utility of ASCII Art in the early computing era is clear. In today's world, where transmission of photograph quality images is not a problem, ASCII art still has relevance. For example, the proposed method may be employed in image manipulation software, or may be used to create ASCII art for the many bulletin board systems that are still popular today,

Finally, in this work I have concentrate on binary images, where the resultant ASCII art is monochromatic. However, it is possible to create multicolor ASCII art, where a binary image is created from a color image and ASCII art conversion is performed giving a monochromatic ASCII art representation, which is subsequently used to mask the original color image.

4 DESIGN METHODOLOGY OF THE PROPOSED ALGORITHM

On designing this algorithm,I have considered that the crypto analyst knows all details of the algorithm. This conforms to "Kickoffs' Principle" in cryptography, which holds that "the security of a cryptographic system should rely only on the key material". The basic idea of our proposed encryption algorithm is hiding a number of bits from plain text message into a random victor of bits. The location of the hiding bits are determined by a pre agreed-upon key by the sender and the receiver. The following subsection gives more details about our algorithm.

Summary of Block-encryption algorithms

	Key Length	Block Length	Problem
DES	56 bits	64 bits	key too small
Khufu	64 bits	64 bits	key too small
REDCO II	160 bits	80 bits	Secure
IDEA	128 bits	64 bits	Patented
Skipjack	80 bits	64 bits	Secret

$$k_{ij} \in \{1,2,3,4,5,6,7,8\} \begin{cases} \forall i = 1,...,L\ ; \mathrm{L} \geq 16 \\ \forall j = 1,2 \end{cases}$$

The method is reasonably simple. We have a key matrix K_{Lx2} where,

$$k_{ij} \in \{1,2,3,4,5,6,7,8\} \begin{cases} \forall i = 1,...,L\ ; \mathrm{L} \geq 16 \\ \forall j = 1,2 \end{cases}$$

This key is known only to the sender and receiver. When the first party wants to send a message M to the second party, he/she determines the key 2 *L K* × and every character from the message is replaced by a binary value. An eight-bit octet is generated randomly and set in a temporary vector V. the bits in the vector V from position K [1,1] to position K[1,2] are replaced by bits from the secret message.

Then the resulting vector V is stored in a file. As long as the message file has not reached its end yet, we move to the next row of the key matrix and another octet is generated randomly and the replacement is performed repeatedly and the resulting vector is stored in the file. The previous procedure is repeated over and over again pending the end the message. The resulting file is sent to the receiver who beforehand has the key matrix. If the key Length is not enough to cover the whole message during the encryption process, the key will be reapplied over and over again until the encryption of the whole message is completed.

5 THE DECRYPTION PROCESS

For decrypting the received encrypted file the following steps are taken. An octet is read from the encrypted binary plain text message EBPM file, then it is set in a temporary vector V, from this vector, bits are extracted from position K(1,1) to position K(1,2) and set in a BPM file. Since the EBPM file is nonetheless not empty, the next octet is read from the EBPM file and then it is set in a temporary vector V. From this vector, bits are extracted from position K (2, 1) to position K (2, 2) and added to the binary plain text message BPM file. The above steps are repeated over and over again until the EBPM file becomes empty. Every octet form the BPM file is transformed to the corresponding character, and then is put in the plaintext file. When the EPBM is empty the plaintext file becomes the message.

In case that the key length is not enough to cover the whole message during the decryption process, the key will be reapplied over and over again till the decryption of the whole message is completed.

6 KEY LENGTH

Now I will show the number of possible keys, i.e., the key space when the key length is 16. The probability of replacing a string of bits whose length ranges from 1 to 8 bit in an octet is 1/64. Consequently, if the key length is 16 there are 6416 = 7.9x1028 possible keys. So we can say that if the attacker has a cipher text and he knows that the key length is 16, there are 7.9x1028 attempts to find the correct key, i.e. , there are 7.9x1028 attempts to find the correct plaintext or secret message.Assuming that a supercomputer working in parallel is able to try 1012 attempts per second, it will take 2.5x109 years to find the secret message. Note that the universe is only 1010 years old. This eliminates brute force attack; however other types of attacks will be discussed in future work.

7 IMPLEMENTATION

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the uses that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation, of change over methods.

1.Testing the developed software with sample data.
2.Debugging of any errors if identified.
3.Creating the files of the system with actual data.
4.Making necessary changes to the system to find out errors.
5.Training of our personnel.

Apart from planning major task of preparing the implementation are education and training of users. The more complex system being implemented, the more involved will be the system analysis and the design effort required just for implementation. On implementation coordinating committee based on policies of individual organization has been appointed.

The implementation process begins with preparing the plan for the implementation for the system. According to this plan, the activities are to be carried out, discussion made regarding the equipment and resources and the additional equipment as to be acquired to implement the new system.

The implementation is the final and important phase. The most critical stage in achieving successful new system and in giving the user confidence that the new system will work

and be effective. The system can be implemented only after thorough testing is done and if it found to working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transaction while using new system.

8 ALGORITHM ANALYSIS

The worst case, regarding storage requirements, occurs when replacing one bit only from message to the V vector. Hence, cipher text equal eight times the size of the plain text. We have analyzed worst case running times for our encryption algorithm and found that it has linear complexity of O(n). Moreover, we have studied the following:
Key length is variable: the key length can be varied from 16 up to any larger value depending on the security level required.

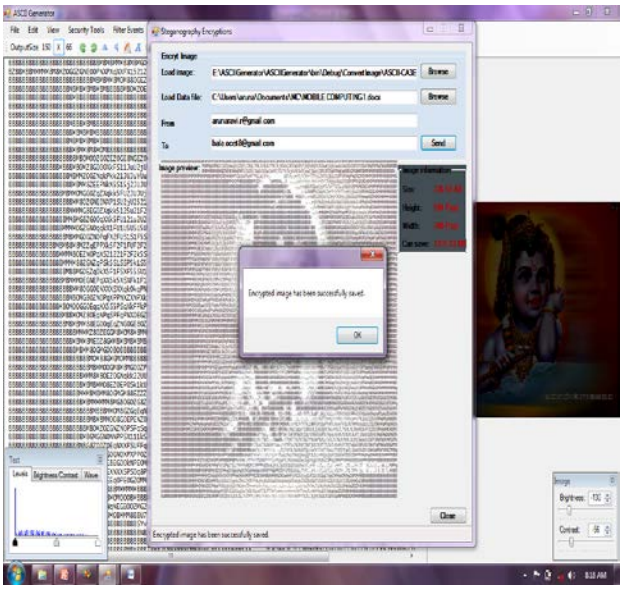
Word length is variable: the block size can be varied between 1 to 16 bits or 1 to 32 bits and so on. That is, encryption can be performed on 16, 32 or 64 bit blocks. This, in turn, can be used on different processor architectures employing 16, 32, or 64 bit registers.The algorithm, therefore, provides variable degrees of security. However, this improved security levels will be at the cost of increased size of the cipher text.

9 OUTPUT EXPERIMENTAL RESULT

9.1 ENCRYPTION PROCESS

The steps involved are,

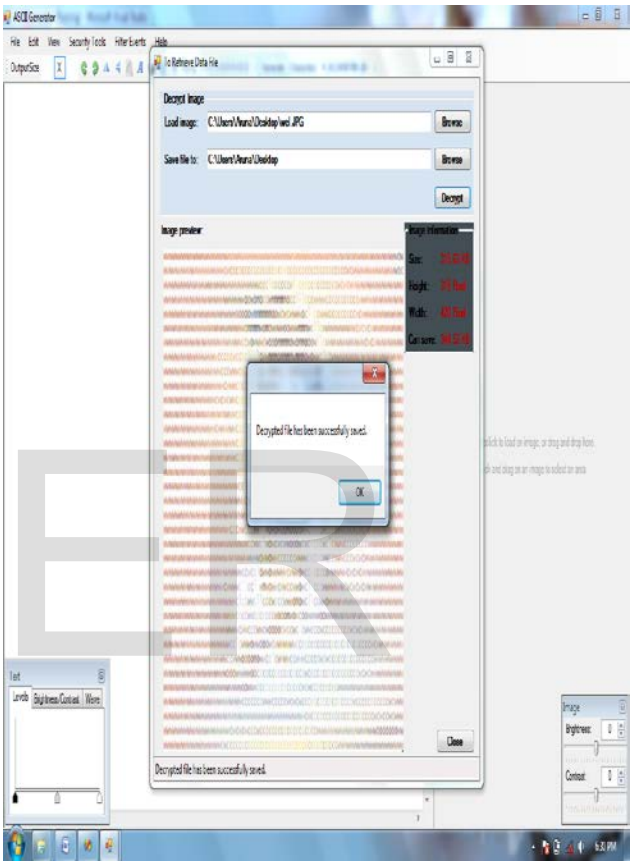
1.Load the converted image.
2.Load the file or image that is to be hided.
3.Encrypt the image.
4.Save the encrypted file.



9.2 DECRYPTION PROCESS

The steps involved are,

1.Load the file saved after encrypted.
2.Assign location to save the decrypted file.
3.Decrypt the file.
4. Original message is separated.



9.3 ENCRYPTED FILE

9.4 DECRYPTED FILE

Digital steganography is the art of inconspicuously hiding data within data. Steganography's goal in general is to hide data well enough that unintended recipients do not suspect the steganographic medium of containing hidden data. As privacy concerns continue to develop along with the digital communication domain, steganography will undoubtedly play a growing role in society. For this reason, it is important that we are aware of digital steganography technology and its implications. Equally important are the ethical concerns of using steganography and stegnoanalysis. Steganography enhances rather than replaces encryption. Messages are not secure simply by virtue of being hidden.

In the computer world, there is a constant struggle to keep secret information secret, private information private, and when profits are involved, protect the copyrights of data. To accomplish these increasingly difficult tasks, new methods based on the principals of stegnography are being developed and used.

10 CONCLUSION

In this application, I have presented a novel application of NMF related methods to the task of automatic ASCII art conversion, where I fit a binary image to a basis constructed from monospace font glyphs using a winner-takes-all assignment. I have presented some examples, and demonstrated that when compared to a standard pseudo inverse approach, non-negative constraints minimize the black space of the ASCII art image, producing better defined curves. Furthermore, I propose the use of the divergence cost function for this task, as it provides an element of control over the final ASCII art representation.

Thus I conclude that the strength of security achieved is very high and unauthorized receiver will not be able to get back the original message using exhaustive without the knowledge of key parameters.Digital Steganography is interesting field and growing rapidly for information hiding in the area of information security. It has a vital role in defense as well as civil applications.

REFERENCES

1.M. N. Schmidt and H. Laurberg. Non-negative matrix factorization with gaussian process priors. Computational Intelligence and Neuroscience, 2008.

2.Amnon Shashua and Tamir Hazan. Non-negative tensor factorization with applications to statistics and computer vision. In ICML '05: Proceedings of the 22nd international conference on Machine learning, pages 792–799, New York, NY, USA, 2005. ACM.

3.Wikipedia. ASCII Art — Wikipedia, the free encyclopedia, 2009. [Online; accessed 11-November-2009].

4.Wikipedia. Unicode — Wikipedia, the free encyclopedia, 2009. [Online; accessed 11- November -2009].

5.Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In Adv. in Neu. Info. Proc. Sys. 13, pages 556–62. MIT Press, 2001.

6.David Guillamet and Jordi Vitria. Classifying faces with non-negative matrix factorization, 2002.

7.Amnon Shashua and Tamir Hazan. Non-negative tensor factorization with applications to statistics and computer vision. In ICML '05: Proceedings of the 22nd international conference on Machine learning, pages 792–799, New York, NY, USA, 2005. ACM.

8. Paris Smaragdis. Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. In Fifth International Conference on Independent Component Analysis, LNCS 3195, pages 494–9, Granada, Spain, September 22–24 2004. Springer-Verlag.

9. D. FitzGerald, M. Cranitch, and E. Coyle. Sound source separation using shifted non-negative tensor factorisation. In Proceedings, IEEE International Conference on Acoustics, Speech and Signal Processing, 2006.

10. Paul D. O'Grady and Barak A. Pearlmutter. Discovering convolutive speech phones using sparseness and non-negativity. In Seventh International Conference on Independent Component Analysis, LNCS 4666, pages 520–7, London, UK, September 2007. Springer-Verlag.