RANDOM THESIS TITLE

A Thesis

Presented to the

Department of Information Systems

and Computer Science

Ateneo de Manila University

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Computer Science

by

Aldrich Ellis C. Asuncion

Brian Christopher T. Guadalupe

2019

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

In this section of the paper, you want to provide the general background and motivation of your study. This can be done defining some key definitions or ideas that govern your study, presenting your context, and narrowing it down with parameters that shall concretize your study. This chapter is divided into four parts: the Context of the Study, the Research Objective/s, the Research Questions, the Scope and Limitations, and finally, the Significance of the Study. Each will be described in greater detail in their respective sections. Appendix A summarizes major points for each of these sections and gives sample guide questions for your reference.

1.1 Context of Study

This subsection of the Introduction aims to discuss the fields related to your study. It is meant to benefit your readers, especially for those who have little or no background on the kind of work your study seeks to explore. You may also briefly discuss related work to help situate your reader. A sample context can be as follows: “The use of agents within ITSs has been the subject of continuing research at the Ateneo de Manila University. A continuum of research work has been invested particularly in the design and development of emotionally intelligent agents for Aplusix. Early work in the area developed models of student emotion [2], [7] as well as preliminary designs for an agent [8]. The first three studies were synthesized into a working ECA [1], with an initial capability of detecting and responding to learner affect through the learner's interaction with Aplusix. (p.2)"

1.2 Research Questions

This subsection mainly presents your research objective/s in question form. An ideal research question must be open-ended, meaning that there is no clear answer that may be readily given to it. The recommended format for this section is that you provide a single overall question, followed by a series of sub-questions (enumerate them for better readability) that help you in answering your main question.

1. Research questions 1...

2. Research questions 2...

3. Research questions 3...

1.3 Research Objectives

With your context established, it is in this section where you will now describe what you intend to achieve by the end of the study. A possible way of presenting this subsection is to first introduce what problem/s you wish to address given the current context, and based on that/these, you present the objectives that would somehow address that/these problem/s. Other common things discussed may include a brief overview of your intended methodology.

1. Research objectives 1...

2. Research objectives 2...

3. Research objectives 3...

1.4 Scope and Limitations of the Study

In order for your study to be realistic, doable, and still produce a concise answer to your research questions within a particular period of time, this section provides you the opportunity to both cross-out extraneous variables, as well as set those aspects that must be controlled in your study. Some common limitation targets include the target audience for the experiment, the software to be used, portability and reusability of the software, the features it will have, and so on.

1.5 Significance of the Study

If the Research Objective answers the ‘what’ of your study, this section aims to answer the ‘why’. Generally, you may address this significance in terms of its significance in the area of Computer Science and the community. However, other questions which you may also use to guide your significance are the following: “Why is your study important?” “What will your study contribute to the field of Computer Science?” “How will your study benefit not only your intended audience, but the general public, especially our country?”

CHAPTER II

REVIEW OF RELATED LITERATURE

2.1 An Introduction to Cryptography and Cryptosystems

In cryptography, a cryptosystem consists of an encryption function $\mathcal{E}$ and a decryption function $\mathcal{D}$, along with the plaintext space $\mathcal{P}$, ciphertext space $\mathcal{C}$ and the key space $\mathcal{K}$ [3]. A *plaintext* is text that can be easily understood by everybody. On the other hand, a *ciphertext* is a result from encrypting the plaintext using an encryption key. The plaintext space consists of all the possible plaintexts.

There are two kinds of cryptosystems, namely: *symmetric* and *asymmetric*. In symmetric encryption, the key is used for both encryption and decryption. On the other hand, asymmetric cryptosystems use separate keys for encryption and decryption. The encryption key (also called the *public key*) is shared to everybody, while the decryption key (also called the *private key*) is kept secret. Because of this, there is no need to agree upon some secure key sharing protocols. Usually, the security of asymmetric cryptosystems relies on the intractability of certain computational problems, like the RSA depends on the difficulty of integer factorization, while ElGamal depends on the difficulty of the discrete logarithmic problem.

A cryptosystem is said to be homomorphic if its encryption function is homomorphic, that is, if it satisfies the relation

$$\mathcal{E}\left(p_1 \otimes p_2\right) = \mathcal{E}\left(p_1\right) \oplus \mathcal{E}\left(p_2\right) \tag{2.1}$$

where $\otimes$ and $\oplus$ are operations in $\mathcal{P}$ and $\mathcal{C}$ respectively [1].

2.2 Common Image Operations

In image processing, the typical image operations being done are intensity transformations which maps intensity values to another, and the use of spatial filters to do operations such as edge detection and image blurring.

2.2.1 Intensity Transformation

Intensity transformations are typically point operations, where a certain operation is applied to each single pixel of the image. Usually, there is a function T that maps a pixel value r into a new value r', thus this transformation satisfies the relation $r' = T\left(r\right)$. Examples of intensity transformations are image negation, log transformation, and power-law transformation.

Image negation is an example of an intensity transformation, where the resulting image would be similar to a photographic negative [2]. In this case, suppose the intensity levels of an image are within the range $[0, L-1]$, then image negation can be expressed by

$$T\left(r\right) = L - 1 - r \tag{2.2}$$

The log transformation is used to enhance dark pixels or increase the dark details of an image by mapping low intensity values to a wider range of values [2]. This has the general form

$$T(r) = c \log(1 + r) \tag{2.3}$$

where c is a constant and $r \geq 0$.

The power-law transformation is a family of transformations that have the form

$$T(r) = cr^{\gamma} \tag{2.4}$$

where $c > 0$ and $\gamma > 0$. This is especially useful since many output devices such as printers and display devices follow the power law, and so correcting the power-law response on these devices in a process called *gamma correction* ensures reproducibility and accuracy of images being displayed [2].

2.2.2 Edge Detection and Spatial Filters

Edge detection is used to find and determine the boundaries in an image, commonly used in applications such as image segmentation and feature extraction. This works by detecting so-called *edges*, areas that have abrupt changes in intensity. Edge detection is usually done by using gradient operators that detect such abrupt changes. These operators are commonly known as *spatial filter*, which are usually of 3×3 size. A common example of spatial filters is the Sobel operator, with two matrices (also called as kernels) g_x and g_y representing the horizontal and vertical components respectively.

$$g_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad \text{and} \qquad g_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{2.5}$$

To get the resulting image I', a convolution is performed between the original image I of size $M \times N$ and the kernel k of size $m \times n$. Now suppose that the pixel value of an image at point (i, j) is $r_{i,j}$. Then, a transformation using spatial filters can be described as follows:

$$\begin{aligned} T\left(r_{i,j}\right) &= [k * I]\left(\left\lfloor \frac{m}{2} \right\rfloor, \left\lfloor \frac{n}{2} \right\rfloor\right) && (2.6) \\ &= \sum_{u=1}^{m} \sum_{v=1}^{n} \left[k_{i,j} r_{i+u,j+v}\right] && (2.7) \end{aligned}$$

Spatial filters are not only used for edge detection, but there are filters that do image blurring (such as Gaussian blur and box blur) and image sharpening.

2.3 Related Work and Previous Implementations

There has been work done regarding the implementation of CryptoImg

example of homomorphic encryption / image manipulation past work

minor limitation: improvement on previous work, but not a direct comparison major limitation: does not discuss security: are modified images also secure?

HElib

2.4 Summary

CHAPTER III

METHODOLOGY

After introducing your topic of choice, discussing and relating previous work with your own, and presenting the underlying concepts that your study will be working with, this section will enable you to go into fine detail into how you will go about your study. Essentially, whatever data you need to gather, as well as how you will intend to gather them, should be presented here. In order to check whether or not your methodology is sound, two main questions should be answered:

1. Is your methodology replicable?
2. Is your methodology realistic and time-bound?

3.1 Methodology as Replicable

A replicable methodology basically means that anyone who reads your methodology and intends to recreate your study to the letter must be able to obtain a similar, if not, exactly the same set of results. It is important, therefore, that you be as specific as you can when describing your methods, such as properly delineating your study's independent, dependent, and control variables. Much like in the literature review and framework, it is good practice to organize your methodology into subsections for easier readability. Of course, apart from generating data given these variables, included in making the methodology replicable is providing the users an effective and appropriate means to collect data for analysis later on. This assumes, of course, that the data you intend to collect is actually measurable, whether it be quantitative (numerical) or qualitative (descriptive).

3.2 Methodology as Realistic and Time-Bound

On the other hand, a realistic and time-bound methodology takes into consideration the context of the researcher. Although a high-level of competency is expected from a graduating CS major, one must also ensure that the proposed study's level of difficulty is aligned with what limited resources is available, especially time. In fact, given that the trend is that you will undergo actual implementation only after being able to defend your proposal during the first semester, the study should be accomplishable at the most within only a semester. It is therefore imperative in the methodology, especially in its initial presentation during the defense, that the timetable for the study is thoroughly laid out, with workable time frames and specific dates for deliverables.

3.3 Summary and Additional Guide Questions

The methodology, in summary, is your detailed explanation of how you intend to go about implementing your study.

BIBLIOGRAPHY

[1] FONTAINE, C., AND GALAND, F. A Survey of Homomorphic Encryption for Nonspecialists. *EURASIP Journal on Information Security 2007* (2007), 1–10.

[2] GONZALEZ, R. C., AND WOODS, R. E. *Digital Image Processing*, 3rd ed. Prentice Hall, Upper Saddle River, N.J, 2008.

[3] TILBORG, H. C. A., Ed. *Encyclopedia of Cryptography and Security*. Springer US, 2005.