PERFORMANCE OF LEVENBERG-MARQUARDT ALGORITHM IN FACE RECOGNITION

NORJULIANA BT ABDUL HADI

This report is submitted in partial fulfillment of the requirements for the award of Bachelor of Electronic Engineering (Computer Engineering) With Honours

Faculty of Electronic and Computer Engineering
Universiti Teknikal Malaysia Melaka

APRIL 2009

UNIVERSTI TEKNIKAL MALAYSIA MELAKA
FAKULTI KEJURUTERAAN ELEKTRONIK DAN KEJURUTERAAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : PERFORMANCE OF LEVENBERG-MARQUARDT ALGORITHM IN FACE RECOGNITION.

Sesi Pengajian : 2008/2009

Saya NORJULIANA BT ABDUL HADI
(HURUF BESAR)

mengaku membenarkan Laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (√) :

[/]	**SULIT***	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
[ ]	**TERHAD***	(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
[ ]	**TIDAK TERHAD**	

Disahkan oleh:

___________________________ (TANDATANGAN PENULIS)

___________________________ (COP DAN TANDATANGAN PENYELIA)

Alamat Tetap: No:21, Lorong Pasdec Makmur 2,
Taman Pasdec Makmur,
26800 Kuala Rompin,
Pahang Darul Makmur.

Tarikh: 30 April 2009

Tarikh: 30 April 2009.

“I hereby declare that this report is the result of my own work except for quotes as cited in the references.”

Signature	**:**
Author	**: Norjuliana Bt. Abdul Hadi**
Date	**: 30 April 2009**

"I hereby declare that I have read this report and in my opinion this report is sufficient in terms of the scope and quality for the award of Bachelor of Electronic Engineering (Computer Engineering) With Honours."

Signature	**:**
Supervisor's Name	**: Puan Norazlina Bt. Abd. Razak**
Date	**: 30 April 2009**

Specially dedicated to my family for their supports and eternal love.

ACKNOWLEDGEMENT

Praise to Allah, the Most Gracious and Most Merciful, who has created the mankind with knowledge, wisdom and power.

First of all, I would like to express this deepest gratitude to my project supervisor, Puan Norazlina Bt. Abd. Razak who has been there to provide continuous guidance, advice, encouragement, support and generous amount of time in helping me to complete this project. I would not have completed this project successfully without her assistance.

Also thankful to my beloved mum and dad who have been very supportive throughout the project and giving me the encouragement all the time. Their understanding and support in term of moral and financial were entirely significance towards the project completion.

Appreciation is also acknowledged to all my fellow friends those who have contributed directly or indirectly in the completion of this project. Their willingness to help, opinions and suggestion on some matters, advices and technical knowledge are simply precious while doing upon completion of my final year project.

ABSTRACT

This thesis investigates the performance of Levenberg-Marquardt Algorithm in face recognition. This project aim to analyze the performance of the algorithm in face recognition system. The Levenberg-Marquardt algorithm is an iterative technique that locates the minimum of function that is expressed as the sum of square of nonlinear functions. It has become a standard technique for nonlinear least-square problems and can be thought of as a combination of steepest descent and the Gauss-Newton method. This project has focused on using the neural network approach and image processing. The Matlab software has been chosen as programming software because it has an image processing toolbox robot vision and neural network toolbox. The Matlab's m-files describe a neural network which will identify if an image contains a human face. For this project, 10 difference pictures (only face) from each 10 peoples will add to database. By using Matlab software, related programs will be saving in m-files. The neural network will train for learning and collecting data process. Then, neural network must be testing for recognizing and collecting data. To train the neural network, only 5 from 10 pictures for each people will be used while the other 5 pictures will be used to test the performance of the neural netwok. Time and performance for training and testing process must be recorded to analyze the data. With respect to database search system, Levenberg-Marquardt Algorithm is not so good in terms of training time. Minimization of the memory of image might be a good solution. Moreover the processing time is also influenced by the number of images trained. If the number of images more, so time taken for training process will be more than before. The objectives of this project have been achieved. For Levenberg-Marquardt algorithm, it can recognize 45 of 50 tested images, which mean 90% of the testing image can be recognized.

ABSTRAK

Tesis ini mengkaji prestasi Algoritma Levenberg-Marquardt dalam proses pengenalan muka. Matlamat utama projek ini adalah untuk menganalisis prestasi algoritma dalam sistem pengenalan muka. Algoritma Levenberg-Marquardt adalah satu teknik yang mengesan fungsi minima yang dinyatakan sebagai fungsi-fungsi jumlah segi empat tak linear. Ia telah menjadi satu teknik yang baik untuk masalah-masalah terkecil tak linear kuasa dua dan boleh difikirkan sebagai satu gabungan penurunan tercuram dan kaedah Gauss-Newton. Projek ini adalah tertumpu dengan menggunakan pendekatan jaringan saraf dan pemprosesan imej. Perisian Matlab telah dipilih sebagai perisian pengaturcaraan kerana ia mempunyai satu peti alat pemprosesan imej wawasan robot dan peti alat jaringan saraf. Untuk projek ini, 10 gambar mimik muka yang berbeza-beza (muka sahaja/saiz pasport) daripada setiap 10 orang akan dimasukkan kepada pangkalan data. Dengan menggunakan perisian Matlab, program-program yang terlibat akan disimpan dalam m-fail. Jaringan saraf akan dilatih untuk sesi pembelajaran dan masa yang diambil direkodkan untuk dianalisis. Kemudian, jaringan saraf mestilah diuji untuk proses pengecaman dan masa yang diperoleh direkodkan untuk dianalisis. Untuk melatih jaringan saraf, hanya 5 daripada 10 gambar-gambar bagi tiap-tiap orang akan digunakan untuk sesi pembelajaran manakala 5 gambar yang lain akan digunakan untuk menguji jaringan saraf tersebut. Masa yang diambil dan prestasi untuk melatih dan menguji jaringan saraf tersebut direkodkan untuk proses penganalisaan data. Algoritma Levenberg-Marquardt adalah tidak bagus dari segi masa persembahan semasa proses melatih jaringan saraf. Mengecilkan saiz imej merupakan salah satu cara penyelesaian. Walau bagaimanapun, objektif projek ini adalah tercapai apabila 90% imej telah berjaya dikenalpasti semasa proses pengecaman imej-imej.

TABLE OF CONTENTS

II LITERATURE REVIEW

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDIX

CHAPTER I

INTRODUCTION

1.1 Overview of the Levenberg-Marquardt Algorithm

Faces have been used by humans to recognize each other for thousands of years. With the increased need for fast and reliable authentication systems, the used advanced automated systems based on biometric technology become necessity and this is now growing rapidly. This project is proposed in order to see the performance of Levenberg-Marquardt (LM) algorithm for face recognition system. The factors which are going to affect the recognition are variations in pose, light intensity and expression of faces.

The Levenberg-Marquardt (LM) algorithm is an iterative technique that locates the minimum of a multivariate function that is expressed as the sum of squares of non-linear real-valued functions. It has become a standard technique for non-linear least-squares problems, widely adopted in a broad spectrum of disciplines. Levenberg-Marquardt (LM) can be thought of as a combination of steepest descent and the Gauss-Newton method. When the current solution is far from the correct one, the algorithm behaves like a steepest descent method: slow, but guaranteed to 1 converge. When the current solution is close to the correct solution, it becomes a Gauss-Newton method.

1.2 Objective of the Project

The Levenberg-Marquardt algorithm is an iterative technique that locates the minimum of function that is expressed as the sum of square of nonlinear functions. It has become a standard technique for nonlinear least-square problems and can be thought of as a combination of steepest descent and the Gauss-Newton method.

The aim of this project is to analyze the performance of the Levenberg-Marquardt algorithm in face recognition system by comparing the performance with the other algorithms like Resilient Backpropagation algorithm, and Scaled Conjugate algorithm at the end of this project.

1.3 Problem Statement

Human face recognition is a difficult problem in computer vision. Early artificial vision experiments tended to center around toy problems in which the world being observed was carefully controlled and constructed. Face recognition is challenging because it is a real world problem. The human face is a complex, natural object that tends not to have easily (automatically) identified edges and features. Because of this, it is difficult to develop a mathematical model of the face that can be used as prior knowledge when analyzing a particular image.

1.4 Scope of the Project

This project was divided into one stage, which is Software Development.

Stage 1: Software Development.

Firstly, literature studies on the concept of Levenberg-Marquardt algorithm, neural network approach and image processing. The Matlab software has been chosen as programming software because it has an image processing toolbox robot vision and neural network toolbox. 10 different pictures (passport size) from each 50 peoples were added to database and from the testing session we analyze the performance results. Time taken for algorithm to recognize the face image must be observed.

1.5 Project Planning

This project is implemented base on the project planning schedule. The project started from July 2008 to April 2009.

1.6 Thesis Outline

Chapter 1 presents an overview to Levenberg-Marquardt algorithm, the objective of the project, scope of the project, project schedule, and thesis outline. Chapter 2 covers the literature review on the Levenberg-Marquardt algorithm, neural network, and image processing. Chapter 3 will describes in details face recognition method. Chapter 4 is covers for results and discussion that was obtained from this project. Then, Chapter 5 presents conclusion and future development for this project.

CHAPTER II

LITERATURE REVIEW

2.1 Introduction

The main objective of this chapter is to review the literature regarding the concept of Levenberg-Marquardt algorithm, neural network approach and image processing.

2.2 Levenberg-Marquardt Algorithm

The Levenberg-Marquardt algorithm is the most widely used optimization algorithm. It outperforms simple gradient descent and other conjugate gradient methods in a wide variety of problems. This document aims to provide an intuitive explanation for this algorithm. The Levenberg-Marquardt algorithm is first shown to be a blend of vanilla gradient descent and Gauss-Newton iteration. Subsequently, another perspective on the algorithm is provided by considering it as a trust-region method.

The problem for which the Levenberg-Marquardt algorithm provides a solution is called Nonlinear Least Squares Minimization. This implies that the function to be minimized is of the following special form :

$$f(x) = \frac{1}{2}\sum_{j=1}^{m} r_j^2\,(x)$$

where $x = (x_1, x_2, \dots, x_n)$ is a vector, and each r_j is a function from $\Re^n$ to $\Re$. The r_j are referred to as a residuals and it is assumed that $m \geq n$.

To make matters easier, f is represented as a residual vector $r: \Re^n \rightarrow \Re$ defined by:-

$$r(\mathrm{x}) = \left(\mathrm{r_1(x)}, \mathrm{r_2(x)}, \dots, \mathrm{r_m(x)}\right)$$

Now, f can be rewritten as $f(x) = \frac{1}{2}\|r(x)\|^2$. The derivatives of f can be written using the Jacobian matrix J of r w.r.t x defined as:-

$$J(x) = \frac{\partial_{rj}}{\partial_{xi}}, 1 \leq j \leq m, 1 \leq i \leq n$$

Let us first consider the linear case where every ri function is linear. Here, the Jacobian is constant and we can represent r as a hyperplane through space, so that f is given by the quadratic $f(x) = \frac{1}{2}\|Jx + r(0)\|^2$. We also get$\nabla f(x) = \mathrm{J}^{\mathrm{T}}\,(\mathrm{Jx} + \mathrm{r})$ and $\nabla^2 f(x) = J^T J$. Solving for the minimum by setting $\nabla f(x) = 0$, we obtain $x_{min} = -(J^T J)^{-1} J^T r$, which is the solution to the set of normal equations.

Returning to the general, non-linear case, we have:-

$$\nabla f(x) = \textstyle\sum_{j=1}^{m} r_j(x)\,\nabla r_j(x) = J(x)^T r(x) \qquad (1)$$

$$\nabla^2 f(x) = J(x)^T J(x) + \textstyle\sum_{j=1}^{m} r_j(x)\,\nabla^2 r_j(x) \qquad (2)$$

The distinctive property of least-squares problems is that given the Jacobian matrix J, we can essentially get the Hessian $\left(\nabla^2 f(x)\right)$ for free if it is possible to approximate the $r_j s$ by linear functions ($\nabla^2 r_j(x)$ are small) or the residuals ($r_j(x)$) themselves are small. The Hessian in this case simply becomes

$$\nabla^2 f(x) = J(x)^T J(x) \tag{3}$$

which is the same as for the linear case.

The common approximation used here is one of near-linearity of the ris near the solution so that $\nabla^2 r_j(x)$ are small. It is also important to note that (3) is only valid if the residuals are small. Large residual problems cannot be solved using the quadratic approximation, and consequently, the performance of the algorithms presented in this document is poor in such cases.

2.2.1 LM as a blend of Gradient descent and Gauss-Newton iteration

Vanilla gradient descent is the simplest, most intuitive technique to find minima in a function. Parameter updation is performed by adding the negative of the scaled gradient at each step, i.e.

$$x_{i+1} = x_i - \lambda \nabla f \tag{4}$$

Simple gradient descent suffers from various convergence problems. Logically, we would like to take large steps down the gradient at locations where the gradient is small (the slope is gentle) and conversely, take small steps when the gradient is large, so as not to rattle out of the minima. With the above update rule, we do just the opposite of this. Another issue is that the curvature of the error surface may not be the same in all directions. For example, if there is a long and narrow valley in the error surface, the component of the gradient in the direction that points along the base of the valley is very small while the component along the valley walls is quite large. This results in motion more in the direction of the walls even though we have to move a long distance along the base and a small distance along the walls.

This situation can be improved upon by using curvature as well as gradient information, namely second derivatives. One way to do this is to use Newton's method to solve the equation $\nabla f(x) = 0$. Expanding the gradient of f using a Taylor series around the current state x_0, we get:-

$$\nabla f(x) = \nabla f(x_o) + (x - x_0)^T \nabla^2 f(x_0) + \textit{higher order terms of } (x - x_0) \tag{5}$$

If we neglect the higher order terms (assuming f to be quadratic around x_o), and solve for the minimum x by setting the left hand side of (5) to 0, we get the update rule for Newton's method –

$$x_{i+1} = x_i - \left(\nabla^2 f(x_1)\right)^{-1} \nabla f \nabla(x_i) \tag{6}$$

where x_0 has been replaced by x_i and x by x_{i+1}.

Since Newton's method implicitly uses a quadratic assumption on f (arising from the neglect of higher order terms in a Taylor series expansion of , the Hessian need not be evaluated exactly. Rather the approximation of (3) can be used. The main advantage of this technique is rapid convergence. However, the rate of convergence is sensitive to the starting location (or more precisely, the linearity around the starting location).

It can be seen that simple gradient descent and Gauss-Newton iteration are complementary in the advantages they provide. Levenberg proposed an algorithm based on this observation, whose update rule is a blend of the above mentioned algorithms and is given as:~

$$x_{i+1} = x_i - (H + \lambda I)^{-1} \nabla f(x_i) \tag{7}$$