



Universidade Federal de Campina Grande

Centro de Engenharia Elétrica e Informática

Curso de Graduação em Engenharia Elétrica

RAJIV ALBINO TORREÃO MOTA

RELATÓRIO DE ESTÁGIO SUPERVISIONADO

Campina Grande, Paraíba
Dezembro de 2015

RAJIV ALBINO TORREÃO MOTA

RELATÓRIO DE ESTÁGIO SUPERVISIONADO

Relatório de Estágio Supervisionado submetido à Unidade Acadêmica de Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Processamento Digital de Imagens

Orientadora:

Professora Luciana Ribeiro Veloso, D. Sc.

Campina Grande, Paraíba
Dezembro de 2015

RAJIV ALBINO TORREÃO MOTA

RELATÓRIO DE ESTÁGIO SUPERVISIONADO

Relatório de Estágio Supervisionado submetido à Unidade Acadêmica de Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Processamento Digital de Imagens

Aprovado em / /

Professor Avaliador

Universidade Federal de Campina Grande
Avaliador

Professora Luciana Ribeiro Veloso, D. Sc.

Universidade Federal de Campina Grande
Orientadora, UFCG

Dedico este trabalho aos meus pais.

AGRADECIMENTOS

Agradeço primeiramente aos meus pais, Carlos Augusto e Maria do Socorro, por terem me proporcionado boa educação.

Agradeço à minha família pelo constante apoio às minhas aventuras ao longo do curso.

Agradeço à minha namorada Aline pelo carinho, companheirismo e amor.

Agradeço aos meus amigos do trabalho, Arnaldo, David, Iron, João, Ramon e Rodrigo pelos ensinamentos e pelas boas risadas.

Agradeço aos funcionários da Vsoft pelo ambiente de trabalho amigável.

Agradeço a Pedro Alves pela oportunidade e confiança.

Agradeço à professora Luciana Veloso pelos conselhos e orientações ao longo deste trabalho.

“A simplicidade é a máxima sofisticação.”

Leonardo da Vinci.

RESUMO

Este relatório descreve as atividades realizadas por Rajiv Albino Torreão Mota, aluno do curso de Engenharia Elétrica da Universidade Federal de Campina Grande, durante o estágio curricular supervisionado de 210 horas, na empresa Vsoft Tecnologia. São reportadas atividades de otimização de algoritmos de processamento digital de imagens utilizados na extração de características de impressões digitais, parte de um sistema biométrico desenvolvido pela empresa em linguagem C++, utilizando a biblioteca de visão computacional OpenCV.

Palavras-chave: processamento digital de imagens, otimização de algoritmos, biometria, impressão digital.

ABSTRACT

This paper reports the activities done by Rajiv Albino Torreão Mota, electrical engineer student at Federal University of Campina Grande, during the supervised internship of 210 hours, in Vsoft Technology Company. Activities related to optimization of image processing code used for fingerprint features extraction, part of a biometric system developed by the company in C++ language, using the computer vision library OpenCV.

Keywords: image processing, code optimization, biometrics, fingerprint.

LISTA DE ILUSTRAÇÕES

Figura 1. Extração de características de impressão digital.	14
Figura 2. Curvas típicas das taxas FAR e FRR.	18
Figura 3. Aplicação do filtro da mediana para remoção de ruído.	19
Figura 4. Diferença entre sucessivas aplicações do fiiltro da mediana (esquerda) e do filtro da moda (direita).	21
Figura 5. Aplicação do filtro LogAbout.	23
Figura 6. Representação espacial do filtro de Gabor.	24
Figura 7. Representação bidimensional do filtro de Gabor.	24
Figura 8. Convolução com filtros de Gabor de diferentes orientações.	24
Figura 9. Etapas do processo de extração de minúcias.	26

LISTA DE TABELAS

Tabela 1. Tempo médio de execução do filtro da mediana.	20
Tabela 2. Tempo médio de execução do filtro da moda.	22
Tabela 3. Tempo médio de execução do filtro LogAbout.	23
Tabela 4. Tempo médio de execução da convolução.	25
Tabela 5. Tempo médio de execução da extração completa.	26
Tabela 6. Resultados do filtro da mediana.	28
Tabela 7. Resultados do filtro da moda.	29
Tabela 8. Resultados do filtro do LogAbout.	29
Tabela 9. Resultados da convolução.	30
Tabela 10. Resultado final da otimização da extração de minúcias.	31

LISTA DE ABREVIATURAS E SIGLAS

DFT	<i>Discrete Fourier Transform</i> (Transformada Discreta de Fourier)
EER	<i>Equal Error Rate</i> (Taxa de Erro Igual)
FFT	<i>Fast Fourier Transform</i> (Transformada Rápida de Fourier)
FVC	<i>Fingerprint Verification Competition</i>
IDFT	<i>Inverse Discrete Fourier Transform</i> (Transformada Inversa Discreta de Fourier)
LUT	<i>Look-Up Table</i> (Tabela de Pesquisa)
PDI	Processamento Digital de Imagens
RAM	<i>Random Access Memory</i> (Memória de Acesso Aleatório)

SUMÁRIO

1	Introdução.....	13
1.1	Objetivos.....	14
1.2	Objetivos Específicos.....	15
1.3	Empresa	15
2	Fundamentação Teórica.....	16
3	Atividades Desenvolvidas	19
3.1	Otimização de Filtros Básicos.....	19
3.1.1	Filtro da Mediana.....	19
3.1.2	Filtro da Moda	21
3.1.3	Filtro LogAbout.....	22
3.2	Convolução com Filtros de Gabor	23
3.3	Melhorias Gerais na Extração de Minúcias	26
4	Resultados	28
4.1	Otimização de Filtros.....	28
4.1.1	Filtro da Mediana.....	28
4.1.2	Filtro da Moda	29
4.1.3	Filtro LogAbout.....	29
4.2	Convolução com Filtros de Gabor	30
4.3	Melhorias Gerais na Extração de Minúcias	30
5	Conclusão	32
	Bibliografia.....	33

1 INTRODUÇÃO

Este trabalho tem a finalidade de relatar as atividades desenvolvidas durante a realização do estágio supervisionado do aluno Rajiv Albino Torreão Mota na empresa Vsoft Tecnologia, na área de processamento digital de imagens aplicado a biometria.

A Visão Computacional permite aos sistemas digitais extraírem informações de imagens. As informações extraídas podem ser utilizadas para diversos fins, entre eles está o reconhecimento de padrões complexos, como texturas, objetos, textos, padrões biométricos, entre outros (Jain, Hong e Pankanti, 2000). Sistemas de reconhecimento biométricos que utilizam características humanas singulares, como impressões digitais, íris, voz e face, permitindo a diferenciação entre seres humanos (Zhao, Chellappa, *et al.*, 2003), são exemplos de sistemas que utilizam técnicas de Processamento Digital de Imagens.

A proliferação de serviços que necessitam de autenticação gerou uma demanda por novos métodos para estabelecer a identidade dos usuários. Métodos tradicionais para estabelecer a identidade de um usuário incluem mecanismos baseados em conhecimento (por exemplo, senhas) e mecanismos baseados em *tokens* (por exemplo, cartões de identidade). Porém, tais mecanismos podem ser perdidos, roubados ou até manipulados com o objetivo de burlar sistemas. Neste contexto, a verificação e a identificação por biometria surgem como alternativas (Ross, Nandakumar e Jain, 2006).

A autenticação biométrica oferece um mecanismo mais confiável utilizando traços físicos (como a íris ou impressão digital) ou comportamentais (como o modo de escrever) que permitem identificar usuários baseados em suas características naturais. Assim, é possível estabelecer a identidade de um usuário com base em quem ele é ao invés do que ele possui ou do que ele lembra. Dentre os diversos traços biométricos físicos, o reconhecimento biométrico baseado na impressão digital é um dos menos intrusivos (Andreza, 2015).

É importante diferenciar identificação de verificação de indivíduos por impressões digitais. A identificação consiste em checar se uma impressão digital está presente em um banco de imagens pré-rotuladas, através de uma comparação 1:N. A verificação consiste em analisar duas impressões digitais e estabelecer uma

probabilidade de ambas pertencerem ao mesmo dedo, é uma comparação 1:1 (Almeida, 2015).

Neste contexto, este trabalho propõe otimizações nos algoritmos de processamento de imagens que compõem um sistema biométrico de identificação de impressões digitais desenvolvido pela empresa Vsoft Tecnologia.

1.1 OBJETIVOS

O processo de verificação da identidade de um indivíduo pela sua impressão digital é dividido em duas grandes etapas: extração e casamento. A etapa de extração visa padronizar a imagem do dedo destacando pontos específicos, chamados minúcias, que serão comparados com o banco de dados previamente extraído na etapa de casamento. Visualmente, a extração executa o processamento ilustrado na Figura 1, onde os círculos vermelhos são as minúcias extraídas:

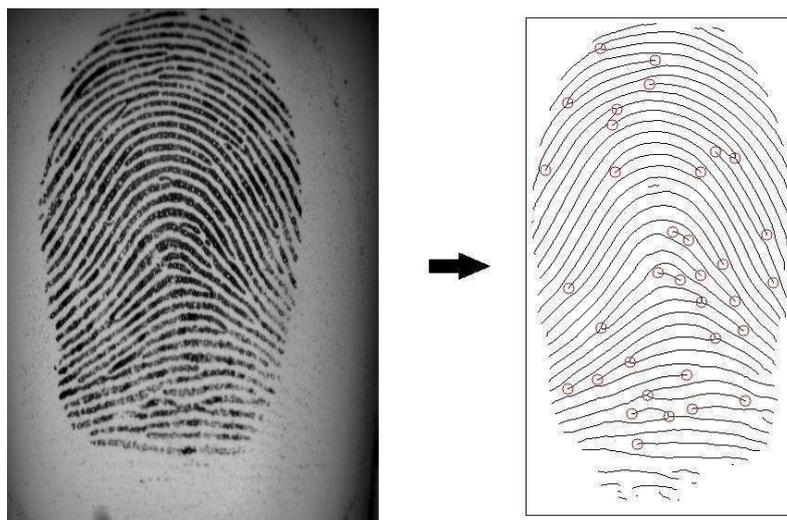


Figura 1. Extração de características de impressão digital.

Esta operação demanda muito esforço computacional, que acaba por deixar o processo lento. O objetivo deste estágio foi reduzir o tempo de processamento da extração de características das imagens mantendo os resultados anteriores alcançados pela empresa.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos deste trabalho são:

- Calcular o tempo de processamento dos algoritmos em questão;
- Otimizar filtros básicos de Processamento Digital de Imagens;
- Elaborar um algoritmo para convolução de imagens no domínio da frequência;
- Otimizar a função de realce de impressões digitais;
- Otimizar a extração de minúcias de impressões digitais;
- Manter as taxas de erro previamente obtidas com o sistema não-otimizado, ou pouco alteradas.

1.3 EMPRESA

A Vsoft Tecnologia é uma empresa que atua no segmento de software, como fornecedora independente e especializada em identificação biométrica.

Sediada em João Pessoa – Paraíba, a Vsoft foi constituída no ano 2000 e se especializou ao longo dos anos no desenvolvimento de software voltado ao processamento digital de imagens. A biometria tem sido seu foco principal, com a pesquisa e desenvolvimento de algoritmos de reconhecimento de faces e impressões digitais.

Em parceria com o Visio (Laboratório de Visão Computacional) da UFPB, a Vsoft vem trabalhando continuamente na pesquisa e desenvolvimento da tecnologia de reconhecimento de faces e impressões digitais BioPass.

O BioPass é testado no FVC (*Fingerprint Verification Competition*), uma das mais importantes avaliações do mercado internacional em biometria, mantida pelo *Biometric System Laboratory* da Universidade de Bolonha (*Università di Bologna*), Itália. Também já teve um de seus módulos certificado pelo órgão americano FBI (*Federal Bureau of Investigation*).

2 FUNDAMENTAÇÃO TEÓRICA

São apresentados os principais conceitos teóricos empregados na realização deste trabalho. Primeiramente, conceitos acerca do processamento digital de imagens e suas vertentes, logo após, definições a respeito de biometria e por fim, as técnicas de medição de tempo de processamento e taxas de erro.

Imagens são sinais, ou seja, funções que conduzem alguma informação a respeito de algo com um interesse (Gonzalez e Woods, 2000). Uma imagem monocromática é uma função $f(x,y)$ onde x e y representam as coordenadas espaciais e o valor de $f(x,y)$ representa um valor de intensidade luminosa do ponto, geralmente chamada nível de cinza. Para serem representadas no computador, é necessário digitalizar as imagens o que pode ser feito através dos processos de amostragem e quantização (Batista, 2005).

O *Processamento Digital de Imagens* (PDI) é o campo da Ciência da Computação que se dedica ao processamento de imagens digitais em um computador digital (Gonzalez e Woods, 2000). Um dos passos fundamentais em PDI é o realce, que tem como finalidade manipular a imagem de forma que o resultado seja mais adequado do que o original para uma aplicação específica.

Uma operação importante na área do PDI é a convolução da imagem com uma pequena matriz, chamada máscara ou janela, usada para aplicar embaçamento, detectar bordas, aplicar textura, remover ruído e diversas outras aplicações (Batista, 2005).

O termo Biometria se refere ao uso de características físicas ou comportamentais, tais como face, íris, impressão digital e voz, para identificar pessoas automaticamente. Conforme (Clarke, 1994), qualquer dessas características humanas pode ser usada como característica biométrica desde que ela satisfaça alguns requisitos básicos:

- Universalidade: toda a população (a ser autenticada) deve possuir a característica. Na prática, temos pessoas que não possuem impressões digitais, por exemplo, por causa de doenças ou acidentes;
- Unicidade: uma característica biométrica deve ser única para cada indivíduo, ou seja, a possibilidade de pessoas distintas possuírem

características idênticas, deve ser nula ou desprezível. Assim, a altura de uma pessoa não é uma boa característica para autenticação, já que várias pessoas podem possuir a mesma altura. Na prática, as características biométricas podem apresentar maior ou menor grau de unicidade.

- Permanência: a característica deve ser imutável num longo intervalo de tempo. Na prática, existem alterações ocasionadas pelo envelhecimento, pela mudança das condições de saúde ou mesmo emocionais das pessoas e por mudanças nas condições do ambiente de coleta.
- Coleta: a característica tem que ser passível de mensuração por meio de um dispositivo. Na prática, todas as características biométricas utilizadas comercialmente atendem a este requisito.
- Aceitação: a coleta da característica deve ser tolerada pelo indivíduo em questão. Na prática, existem preocupações com higiene, com privacidade e questões culturais que diminuem a aceitação da coleta.

É importante salientar que, na prática, nenhuma característica biométrica atende com perfeição a todos os requisitos de uma característica biométrica ideal (Clarke, 1994).

Outras características estáticas também são utilizadas em menor grau por serem de difícil aferição, como o DNA e o padrão da arcada dentária, ou estão em estágios iniciais de pesquisa, como o formato das orelhas, o padrão vascular da retina, o odor do corpo e o padrão de calor do corpo ou de partes dele.

O desempenho da verificação de sistemas biométricos geralmente é medido utilizando métricas como Taxa de Falsa Aceitação (do inglês, *False Acceptance Rate*, FAR), Taxa de Falsa Rejeição (do inglês, *False Rejection Rate*, FRR) e Taxa de Erro Igual (do inglês, *Equal Error Rate*, EER). Falsa aceitação ocorre quando um usuário não registrado obtém o acesso ao sistema protegido, enquanto falsa rejeição refere-se a quando um usuário registrado não consegue obter o acesso legítimo. A EER é geralmente utilizada para comparar o desempenho de diferentes sistemas biométricos (Zang, 2004) e geralmente é calculada pelo ponto de interseção entre as curvas FAR e FRR, conforme indicado na Figura 2. Essas medidas são inversamente proporcionais à qualidade do desempenho do sistema biométrico.

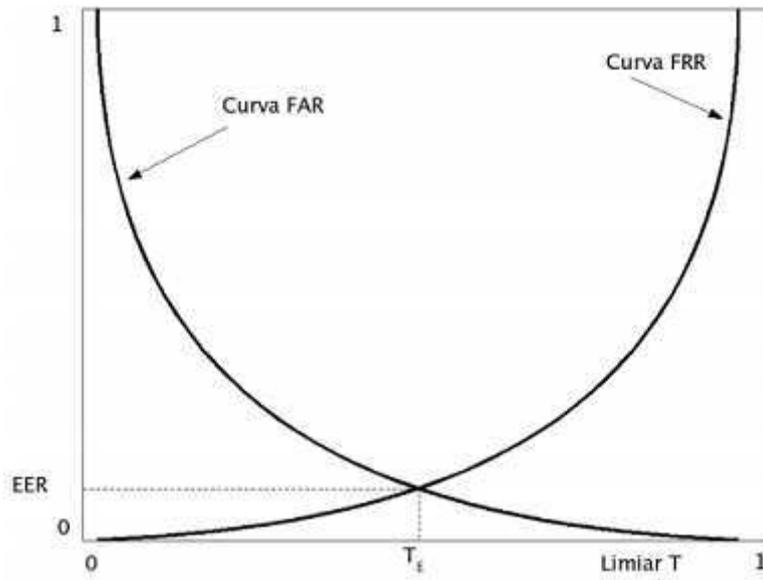


Figura 2. Curvas típicas das taxas FAR e FRR.

Como indicado, o EER é o ponto onde a FAR é igual à FRR.

3 ATIVIDADES DESENVOLVIDAS

As atividades realizadas durante o estágio podem ser divididas em três grupos: otimização de filtros básicos, desenvolvimento de um algoritmo de convolução usando a OpenCV e melhorias gerais no processo de realce.

Todas as medições de tempo de execução dos algoritmos foram realizadas em um computador com processador Intel Core 2 Duo @ 2,8 GHz, com memória RAM de 4 GB, sob o sistema operacional Windows 7/32 bits. Durante as medições, o computador não executava nenhum outro programa paralelamente, apenas processos inerentes ao sistema operacional. Os resultados exibidos são a média de numerosas execuções do mesmo código (a quantidade varia a cada caso), medidas com o comando *Benchmark-Command*, em ShellScript, através do Windows PowerShell.

3.1 OTIMIZAÇÃO DE FILTROS BÁSICOS

Os filtros mais básicos são usados em todos os projetos da empresa, portanto uma otimização nestes ajuda também outros setores. A primeira atividade foi otimizar o Filtro da Mediana, o Filtro da Moda e o Filtro LogAbout.

3.1.1 FILTRO DA MEDIANA

O filtro da mediana é uma técnica não-linear de filtragem bastante utilizado na área de PDI para redução de ruído com preservação das bordas, sob certas condições. Um exemplo desta filtragem com uma janela de dimensões 5x5 é mostrado na Figura 3, onde é usada para remover ruído:



Figura 3. Aplicação do filtro da mediana para remoção de ruído.

Primeiramente é necessário medir o tempo de execução da função. Medindo para cinquenta execuções da função em uma imagem monocromática de dimensões 400×560 e com diferentes dimensões da janela, os tempos médios obtidos estão na Tabela 1:

Tabela 1. Tempo médio de execução do filtro da mediana.

Dimensões da máscara	Tempo médio de execução (s)
3x3	0,0342
5x5	0,1362
7x7	0,3118
9x9	0,5676
11x11	0,9192
13x13	1,3424
15x15	1,9051
17x17	2,5374
19x19	3,2947
21x21	4,1368

Analisando o *loop* principal do filtro, foi detectado que a imagem era percorrida coluna a coluna, e não linha a linha, o que é um erro devido ao modo de funcionamento da memória cache do computador e de como as imagens são armazenadas (linha a linha). Pelo Princípio da Localidade Espacial (Mano e Kime, 2007), a memória cache buscará linhas da memória RAM, acessar apenas um elemento de cada linha fere este princípio e inutiliza o ganho de performance da cache.

Em seguida foi utilizada a ferramenta *Performance Wizard*, do Microsoft Visual Studio 2013, que mede a porcentagem do tempo gasto por cada chamada interna da função, permitindo encontrar gargalos. Esta ferramenta indicou que 84,1% do tempo de execução do algoritmo é gasto em chamadas da função *partial_sort()*, com o objetivo de ordenar um vetor de valores de pixels.

Como esperado, é na ordenação que há um maior consumo de tempo de processamento, visto que esta é uma operação de alta complexidade operacional. Para resolver este problema, foi utilizado um algoritmo de seleção ao invés de uma ordenação. A mediana de um vetor é o elemento central do vetor ordenado, olhando de um diferente ponto de vista, o vetor ordenado tem seus elementos postos em ordem crescente (ou decrescente) e o valor central será o k -ésimo menor elemento onde k é a metade do número de elementos. Sendo assim, é adequado o uso do algoritmo *Quick Select*, que retorna o k -ésimo menor elemento de um vetor. Passando o parâmetro k

como sendo a metade do comprimento, a mediana é encontrada sem fazer ordenação. Em teoria esta é uma otimização válida pois a complexidade do algoritmo *partial_sort* é $O(n \log(n/2))$ no caso médio, enquanto que a do *Quick Select* é $O(n)$ (Arora e Barak, 2007).

Partindo para a otimização do corpo da função, é possível reduzir a quantidade de condições *if/else* na obtenção dos valores dos pixels da máscara. Na varredura da imagem cada pixel da janela é testado para evitar que o código procure pontos fora da imagem, o que ocasionaria um erro de acesso de memória, isto é feito com quatro condições *if* em cada iteração, o que acarreta em um atraso considerável quando levado em consideração todo o processo. Separando a varredura em três partes: borda superior, centro e borda inferior, e cada parte em mais três: borda esquerda, centro e borda direita, há uma intersecção entre as três zonas, o centro, que permite coletar os pixels sem testar as condições de contorno. Como as dimensões da janela são muito menores que as dimensões da imagem, a função passa a maior parte do tempo no *loop* do centro, que sem condições *if/else* é executado mais rápido que as demais zonas. Esta é uma otimização passível de ser usada em muitos outros filtros que usam janelamento, porém deixa o código menos legível.

3.1.2 FILTRO DA MODA

No filtro da moda, o valor do pixel é a moda dos valores dos pixels da imagem em uma vizinhança de n pixels. A moda de um conjunto de valores é o valor mais frequente do conjunto. Assim como o filtro da mediana, também é usado para remoção de ruído, entretanto tem a vantagem de preservar melhor as bordas após sucessivas aplicações, como mostrado na Figura 4:

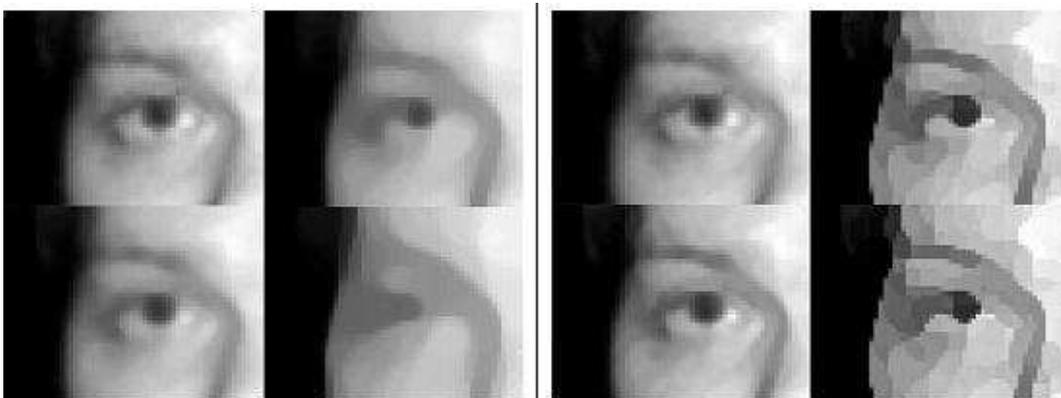


Figura 4. Diferença entre sucessivas aplicações do filtro da mediana (esquerda) e do filtro da moda (direita).

Novamente, foi medido o tempo médio de dez execuções da função para uma imagem monocromática de dimensões 256×384. Os tempos estão na Tabela 2:

Tabela 2. Tempo médio de execução do filtro da moda.

Dimensões da máscara	Tempo médio de execução (s)
3x3	6,45
5x5	11,82
7x7	24,9
9x9	42,57
11x11	65,86

A janela foi limitada a 11x11 devido ao elevado tempo de processamento, e na prática utiliza-se filtro com janelas de tamanho inferiores.

O *Performance Wizard* indicou que 91,1% do tempo de execução do algoritmo consistia em chamadas à função *mode_calc()*, que calcula a moda de um vetor de valores. Esta função conta a incidência de cada pixel na janela e escolhe o maior, esta é uma forma muito trivial de se fazer o cálculo.

O histograma de uma imagem mostra a incidência de cada valor de pixel. A função *calcHist()*, da biblioteca OpenCV, calcula o histograma e retorna um vetor de 256 posições com o número de ocorrências de cada pixel. O maior valor dessa matriz estará na posição do valor que mais se repetiu na janela. Ou seja, a moda é o valor correspondente ao pico do histograma.

Como a função de cálculo do histograma da OpenCV é bastante otimizada, a função *mode_calc()* foi substituída pelo cálculo do histograma dos valores seguido do encontro do valor de pico.

Além desta, as otimizações feitas no filtro da mediana aplicáveis nesta situação também foram implementadas.

3.1.3 FILTRO LOGABOUT

O filtro LogAbout executa a normalização de iluminação de imagens preservando características. É usado para reconhecimento de faces, visto que as diferenças de iluminação dificultam o reconhecimento da identidade de indivíduos devido ao sombreamento em regiões relevantes para a comparação. A Figura 5 ilustra a aplicação do filtro LogAbout:

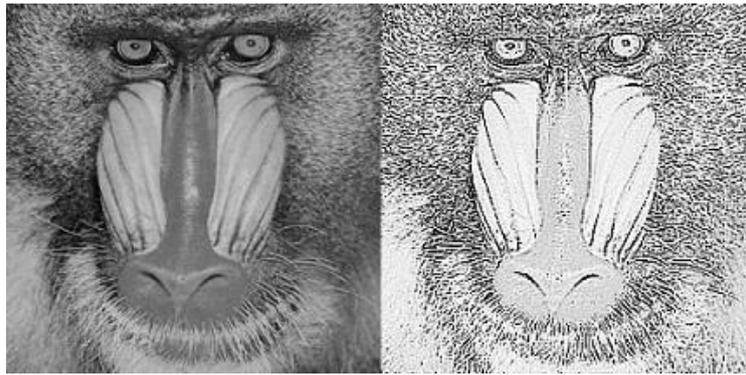


Figura 5. Aplicação do filtro LogAbout.

O tempo de execução para imagens de diferentes dimensões foi medido para cem execuções e suas médias estão dispostas na Tabela 3 em milissegundos:

Tabela 3. Tempo médio de execução do filtro LogAbout.

Dimensões da imagem	Tempo médio de execução (ms)
298×298	3,88
512×619	12,91
5616×2962	1128,46

Primeiramente, era feita uma convolução espacial da imagem com uma máscara, foi trocada por uma convolução utilizando o domínio da frequência, através da função *filter2D()* da OpenCV.

Em seguida, a imagem é percorrida ponto a ponto calculando o logaritmo de cada pixel. Esta operação pode ser otimizada com o uso de uma *Look-Up Table* (LUT), que utiliza uma tabela de 256 posições com o valor do logaritmo de cada possível valor de pixel. Com isto, só é preciso percorrer a imagem atribuindo o novo valor por associação entre o pixel e seu correspondente na tabela, evitando calcular o logaritmo do pixel a cada iteração e economizando processamento.

3.2 CONVOLUÇÃO COM FILTROS DE GABOR

O Filtro de Gabor é um filtro linear usado para detecção de bordas. Ele é seletivo tanto em frequência quanto em orientação e tem resolução conjunta ótima nos domínios espaciais e frequenciais (Maltoni, Maio, *et al.*, 2009). Como mostrado na Figura 6, um filtro de Gabor é definido por uma onda senoidal plana delimitada por uma curva Gaussiana. Na Figura 7 tem-se sua representação bidimensional:

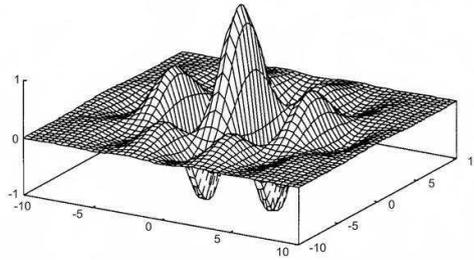


Figura 6. Representação espacial do filtro de Gabor.

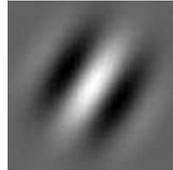


Figura 7. Representação bidimensional do filtro de Gabor.

Um dos métodos mais comuns na literatura para realce de impressões digitais é através da convolução com filtros de Gabor. Utilizando parâmetros coerentes e sucessivas iterações, é possível destacar nitidamente as linhas presentes nas imagens.

Cada convolução com uma máscara de Gabor destaca uma orientação diferente na imagem da digital, variando a orientação do filtro de 0 a π e tratando os resultados é possível destacar todos os contornos. A Figura 8 ilustra este processo, usando um passo de $\pi/12$ radianos:

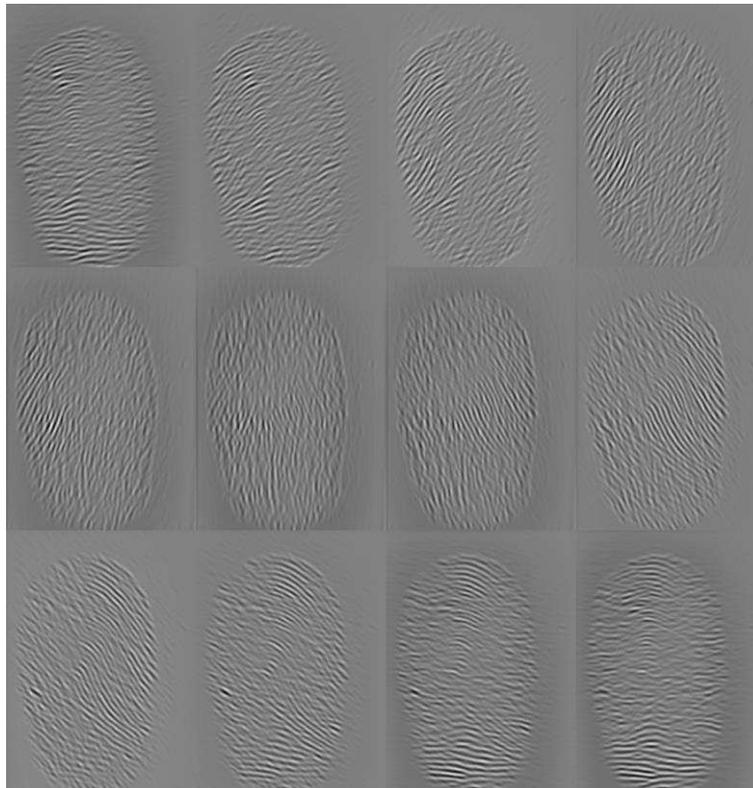


Figura 8. Convolução com filtros de Gabor de diferentes orientações.

Como cada orientação destaca uma área diferente da imagem, é preciso sobrepor os resultados das convoluções para obter uma imagem mais nítida. Este processo demanda um enorme esforço computacional porque é iterativo. Após convoluir a imagem com todos os filtros de Gabor e sobrepor os resultados, a imagem obtida ficará incompleta, fazendo-se necessário repetir o processo até que a imagem esteja 85% completa (esta é a condição de parada), o que acontece em média com 16 iterações.

A função que fazia a convolução no projeto executava os seguintes passos:

- i. Cálculo da DFT da imagem;
- ii. Cálculo da DFT da máscara;
- iii. Multiplicação dos espectros;
- iv. Cálculo da IDFT do resultado.

Os tempos médios foram medidos para cem execuções de cada etapa do processo, os tempos foram os seguintes da Tabela 4 em milissegundos:

Tabela 4. Tempo médio de execução da convolução.

Etapa	Tempo médio de execução (ms)
DFT da imagem	10,120
DFT da máscara	5,139
Multiplicação dos espectros	1,508
IDFT do resultado	11,344
Total	28,111

Há um passo que pode ser descartado: o cálculo da DFT da máscara. As máscaras de Gabor podem ser todas pré-calculadas e guardadas na memória. Portanto, a função foi separada em duas outras, uma para calcular a DFT da entrada e outra para multiplicar os espectros e calcular a transformada inversa.

Todos os cálculos eram feitos com rotinas implementadas no projeto, foi decidido usar as funções da OpenCV por já serem bem otimizadas. Na primeira função, inicialmente o tamanho da imagem é aumentado com as funções *copyMakeBorder()* e *getOptimalDFTSize()* para deixá-la em dimensões ótimas, em seguida é utilizada a função *dft()*. Nenhum tratamento para visualização fez-se necessário neste caso.

No segundo método foram utilizadas as funções *mulSpectrums()*, *idft()* e foi recortado do resultado a imagem nas dimensões da entrada original.

3.3 MELHORIAS GERAIS NA EXTRAÇÃO DE MINÚCIAS

O processo de extração é dividido em seis sub-rotinas. Nas primeiras atividades do estágio o foco era em métodos específicos. Nesta etapa foi necessária uma visão geral sobre o processo para analisar as partes mais significativas do sistema como um todo.

Usando um alto nível de abstração, o processo de extração segue o seguinte algoritmo:

- i. Leitura da imagem;
- ii. Pré-processamento;
- iii. Corte da região de interesse;
- iv. Realce por filtros de Gabor;
- v. Afinamento;
- vi. Localização das minúcias.

A Figura 9 ilustra este processo em respectiva ordem, mostrando que ele é robusto para digitais danificadas, como neste caso por cortes no dedo:

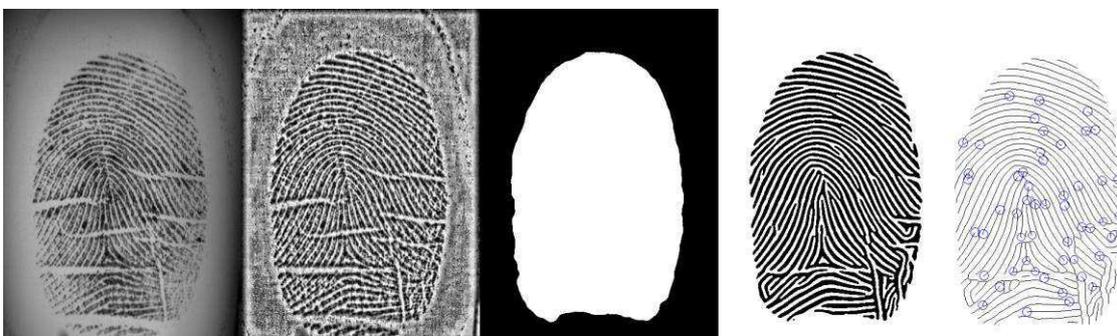


Figura 9. Etapas do processo de extração de minúcias.

Na empresa havia uma tabela de medições de tempo da extração em bancos do FVC, com imagens de diferentes dimensões. As medições estão na Tabela 5.

Tabela 5. Tempo médio de execução da extração completa.

Banco	Dimensões das imagens	Tempo médio de execução (s)
FVC2000Db	448×478	14,747
FVC2002Db	388×374	12,862
FVC2004Db	640×480	14,508
FVC2006Db	400×560	13,344

FVC2006Db	400×560	12,853
FVC2006Dbb	400×560	13,511

Com estes tempos para a extração, o projeto BioPass conseguiu uma taxa de EER (*Equal Error Rate*) de 0,84%, entretanto, o FVC estabelece um tempo máximo de 5 segundos para este processo. Claramente, os tempos medidos estavam bem acima do desejado.

As imagens eram tratadas por classes criadas no projeto BioPass, como *ByteImage** e *FloatImage** por exemplo, e foi necessário criar funções de todos os filtros, transformadas e utilitários para estas classes. Eram muitas classes e funções não otimizadas e com algumas falhas e não se sabia ao certo se havia um bom desempenho para todas. Após testes de performance e discussões sobre a viabilidade de mudar todo o projeto, decidiu-se utilizar as classes da OpenCV, trocando todas as classes de imagens pelo *Mat* da biblioteca, com as seguintes vantagens: gerenciamento automático de memória, facilidade de manipulação dos dados e toda uma gama de funções já implementadas e otimizadas.

Na etapa de pré-processamento há uma equalização e expansão de histograma em blocos, essas funções foram modificadas para utilizar *Mat* e funções da OpenCV.

A aplicação dos filtros de Gabor era feita com 18 passos de orientação e 3 períodos (7, 9 e 11 pixels), totalizando 36 convoluções. Examinando os resultados, as imagens resultantes do período 9 quase não compunham a superposição. Após testar os resultados com apenas os períodos 7 e 11, vimos que a saída era menos de 2% diferente (em porcentagem de pixels brancos e pretos) da anterior, diferença aceitável para o ganho de performance que se consegue reduzindo em um terço o número de convoluções executadas.

Além disso, o tempo de execução do algoritmo da FFT depende das dimensões da imagem. Pelo modo como as multiplicações são rearranjadas, dividindo o cálculo em subgrupos de metade do número de pontos, as imagens com dimensões em potências de 2 são o melhor caso do algoritmo, caso que reduz a complexidade de $O(n^2)$ para $O(n \log n)$ (Arora e Barak, 2007). Aumentando o tamanho das DFTs das imagens para potências de 2 ou próximos, como $2^k \cdot 3$ por exemplo, o tempo do cálculo será reduzido sem alterar o resultado, se este aumento acrescentar apenas zeros. Montando uma tabela de valores ótimos para as dimensões das imagens foi possível reduzir o tempo das FFTs.

No afinamento, apenas a conversão para *Mat*, desmodularização do código e melhores práticas de programação foram suficientes.

4 RESULTADOS

A condição para que uma função seja otimizada é que a nova função tenha exatamente o mesmo resultado e um tempo de processamento menor. Para testar o resultado, a imagem resultante da função original é subtraída da imagem resultante do filtro otimizado, para a mesma entrada, caso a diferença seja uma imagem totalmente preta (uma matriz de zeros), as saídas são iguais. Nos casos das duas primeiras atividades descritas anteriormente estas condições foram satisfeitas. No processo de extração como um todo, houve uma diferença tolerável nos resultados devido à remoção das convoluções com filtros de Gabor de período 9.

4.1 OTIMIZAÇÃO DE FILTROS

Os filtros otimizados tiveram resultados satisfatórios.

4.1.1 FILTRO DA MEDIANA

Após as mudanças descritas, os tempos médios de processamento do filtro nas mesmas condições anteriores estão dispostos na Tabela 6.

Tabela 6. Resultados do filtro da mediana.

Dimensões da máscara	Tempo médio de execução (s)	
	Função Original	Função Otimizada
3x3	0,0342	0,0280
5x5	0,1362	0,0993
7x7	0,3118	0,1860
9x9	0,5676	0,2703
11x11	0,9192	0,3643
13x13	1,3424	0,4838
15x15	1,9051	0,6496
17x17	2,5374	0,7568
19x19	3,2947	0,9139
21x21	4,1368	1,1846

Como o processo de realce chama duas vezes o filtro da mediana com janela 11x11, o tempo total foi reduzido em aproximadamente 1,11 segundos.

4.1.2 FILTRO DA MODA

O quadro comparativo entre os tempos médios originais e após a otimização está na Tabela 7:

Tabela 7. Resultados do filtro da moda.

Dimensões da máscara	Tempo médio de execução (s)	
	Função Original	Função Otimizada
3x3	6,45	3,83
5x5	11,82	3,83
7x7	24,9	3,84
9x9	42,57	3,86
11x11	65,86	3,89

Estes resultados foram melhores do que o esperado. O filtro da moda é usado em outro projeto, segmentação de face, onde o teste em todo o banco levava cerca de doze horas para ser feito por usar o filtro da moda de dimensões 11x11 em imagens tricromáticas. Após esta otimização, o mesmo teste passou a ser realizado em menos de quarenta minutos, o que permitiu realizar muito mais testes em menos tempo e testar pequenas alterações de código.

4.1.3 FILTRO LOGABOUT

Os tempos médios originais foram comparados com os tempos da função otimizados, os resultados estão em milissegundos na Tabela 8:

Tabela 8. Resultados do filtro do LogAbout.

Dimensões da imagem	Tempo médio de execução (ms)	
	Função Original	Função Otimizada
298x298	3,88	2,80
512x619	12,91	7,76
5616x2962	1128,46	433,80

Percentualmente, a melhora de desempenho foi maior na imagem maior devido ao fato das LUTs fazerem mais diferença de processamento em imagens grandes.

4.2 CONVOLUÇÃO COM FILTROS DE GABOR

Os tempos do novo algoritmo de convolução usando a OpenCV foram medidos e comparados com os tempos antigos. A Tabela 9 é o comparativo entre ambos, com tempos em milisegundos:

Tabela 9. Resultados da convolução.

Etapa	Tempo médio de execução (ms)	
	Função Original	Função Otimizada
DFT da imagem	10,120	9,265
DFT da máscara	5,139	-
Multiplicação dos espectros	1,508	1,490
IDFT do resultado	11,344	9,126
Total	28,111	19,881

Este foi um resultado muito importante para o projeto BioPass pois reduziu drasticamente o tempo do realce. A redução foi de aproximadamente 8,23 milisegundos por chamada da convolução, já que as DFTs das máscaras não entram mais no *loop* principal pois são calculadas apenas uma vez. A imagem precisa ser convoluída com cada filtro de Gabor uma vez por iteração, é usado um passo de 10 graus entre as orientações havendo, então, 18 orientações diferentes (0° a 170°) e para cada orientação é preciso usar 2 diferentes períodos, resultando em 24 convoluções por iteração. Este método usa em média 16 iterações para chegar a um resultado aceitável, totalizando 504 convoluções. Multiplicando o valor da redução pelo número de convoluções conclui-se, portanto, que esta otimização reduziu cerca de 3,16 segundos do tempo total do realce.

4.3 MELHORIAS GERAIS NA EXTRAÇÃO DE MINÚCIAS

Finalmente, o processo de extração como um todo teve uma satisfatória redução no tempo de processamento. As medições levam muito tempo para serem feitas pois são passadas em seis bancos de impressões digitais com centenas de imagens cada um, por isso são realizadas medições mensais. Na Tabela 10 estão dispostos os valores da medição de 30 de março, antes do início do trabalho de otimização, junto com a medição do dia 31 de agosto, para fins comparativos.

Tabela 10. Resultado final da otimização da extração de minúcias.

Banco	Dimensões das imagens	Medição 30/03/2015		Medição 31/08/2015	
		Média (s)	Máximo (s)	Média (s)	Máximo (s)
FVC2000Db	448x478	14,747	21,813	3,758	4,781
FVC2002Db	388x374	12,853	17,445	3,314	4,427
FVC2004Db	640x480	14,508	18,968	3,893	4,692
FVC2006Db	400x560	13,344	18,571	3,541	4,595
FVC2006Db _a	400x560	12,862	18,549	3,251	4,628
FVC2006Db _b	400x560	13,511	19,094	3,365	4,624

Como pode ser visto, o objetivo foi alcançado, o tempo da extração está abaixo do limite de 5 segundos exigido pelo FVC em todos os bancos testados, devido ao trabalho de toda a equipe. Em contrapartida, o EER que anteriormente era de 0,84% passou a ser 1,38%. Isto se deve ao fato de que o casamento de impressões digitais é feito com algoritmos genéticos e para esta nova extração é preciso recalibrar os pesos das variáveis, o que futuramente corrigirá este aumento.

5 CONCLUSÃO

No estágio tive a oportunidade de entrar numa área pouco explorada ao longo do curso. O trabalho na Vsoft foi de grande importância para solidificar e expandir meus conhecimentos nas áreas de PDI e programação, alcançando os objetivos com ótimos resultados nas tarefas realizadas.

Foi notória ao longo das atividades a importância das disciplinas Processamento Digital de Sinais e Sistemas de Aquisição de Áudio e Vídeo, que forneceram a base do conhecimento em PDI indispensável neste estágio, bem como as disciplinas Introdução e Técnicas de Programação me ensinaram os fundamentos da programação. As disciplinas Arquitetura de Sistemas Digitais e Cálculo Numérico me ajudaram a entender como funcionam os componentes do computador, conhecimento necessário para aplicar as técnicas de otimização utilizadas.

BIBLIOGRAFIA

- ALMEIDA, I. **Analysis of a Genetic Algorithm Based Approach in the Optimization of the SourceAFI's Matching Algorithm**. International Conference of Scientific Computing, 2015. p. 21-25.
- ANDREZZA, I. L. P. **Análise de técnicas de normalização aplicadas ao reconhecimento facial**. UFPB, João Pessoa, 2015.
- ARORA, S.; BARAK, B. **Computational Complexity: A Modern Approach**. Princeton: Princeton University, 2007.
- BATISTA, L. V. **Notas de Aula da Disciplina Introdução ao Processamento Digital de Imagens**. UFPB. João Pessoa, p. 84. 2005.
- CLARKE, R. **Human Identification in Information Systems: Management Challenges and Public Policy Issues**. 4a. ed. Canberra: Information Technology & People, v. 7, 1994.
- GONZALEZ, R. C.; WOODS, R. E. **Processamento de Imagens Digitais**. 1a. ed. São Paulo: Edgard Blücher LTDA, 2000.
- JAIN, A.; HONG, L.; PANKANTI, S. **Biometric Identification**. New York: ACM, 2000.
- MALTONI, D. et al. **Handbook of Fingerprint Recognition**. 2a. ed. Londres: Springer, 2009.
- MANO, M. M.; KIME, C. R. **Logic and Computer Design Fundamentals**. 4a. ed. Upper Saddle River: Prentice Hall, 2007.
- ROSS, A. A.; NANDAKUMAR, K.; JAIN, A. K. **Handbook of Multibiometrics**. New York: Springer, 2006.
- ZANG, D. D. **Palmprint Authentication**. New York: Springer US, v. III, 2004.
- ZHAO, W. et al. **Face Recognition: A Literature Survey**. ACM Computing Surveys. New York. 2003. p. 95-98.