

UNIVERSIDADE FEDERAL DO PARANÁ

KAUANA GOMES

**RECONHECIMENTO DE INDIVÍDUOS UTILIZANDO TÉCNICAS DE
CLASSIFICAÇÃO DE PADRÕES EM MAPEAMENTO DE VEIAS FACIAIS**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2019

KAUANA GOMES

**RECONHECIMENTO DE INDIVÍDUOS UTILIZANDO TÉCNICAS DE
CLASSIFICAÇÃO DE PADRÕES EM MAPEAMENTO DE VEIAS FACIAIS**

TRABALHO DE CONCLUSÃO DE CURSO

Trabalho apresentado como requisito parcial à
obtenção de nota no Trabalho de Conclusão de
Curso B, do Curso de Engenharia Elétrica com
Ênfase em Sistemas Eletrônicos Embarcados.
Professora Orientadora: Giselle Lopes Ferrari

CURITIBA

2019

LISTA DE ILUSTRAÇÕES

FIGURA 2.1 – PDI na área médica	9
FIGURA 2.2 – Espectro Eletromagnético	10
FIGURA 2.3 – Sistemas biométricos	13
FIGURA 2.4 – Mapeamento de veias dos dedos	14
FIGURA 2.5 – Mapeamento de veias da mão	15
FIGURA 2.6 – Veias da face	16
FIGURA 3.1 – Diagrama de Atividades	17
FIGURA 3.2 – Modelagem de Face	18
FIGURA 3.3 – Código: Código de redimensionamento contido no script	19
FIGURA 3.4 – Terminal - Resposta do compilador ao script de redimensionamento	19
FIGURA 3.5 – Imagem RGB e térmica após script de normalização	20
FIGURA 3.6 – Convolução entre imagem original e kernel	21
FIGURA 3.7 – Filtragem por Gradiente e Filtragem por Médias	21
FIGURA 3.8 – Filtragem por Gradiente e Filtragem por Médias	22
FIGURA 3.9 – Filtragem por Gradiente e Filtragem por Médias	22
FIGURA 3.10 – Imagem Binária Resultante	23
FIGURA 3.11 – Operação de erosão	24
FIGURA 3.12 – Operação de dilatação	24
FIGURA 3.13 – Operação de abertura	25
FIGURA 3.14 – Operação de fechamento	26
FIGURA 3.15 – Operações Morfológicas e Máscara Binária da Imagem	28
FIGURA 3.16 – Operações Morfológicas e Máscara Binária da Imagem	29
FIGURA 3.17 – Operações Morfológicas e Máscara Binária da Imagem	30
FIGURA 3.18 – Gradiente Orientado	30
FIGURA 4.1 – Exemplo de Gradiente Orientado	32
FIGURA 4.2 – Gradiente Orientado	32
FIGURA 4.3 – Gradiente Orientado - Aproximado	33
FIGURA 4.4 – Algoritmo de Vetores de Suporte	34

FIGURA 5.1 – Imagem Binária Inconclusiva	35
FIGURA 5.2 – Imagem Binária Inconclusiva	35
FIGURA 5.3 – Detecção de Veias - Homem A.	36
FIGURA 5.4 – Detecção de Veias - Homem B.	36

SUMÁRIO

1	INTRODUÇÃO	6
1.1	DEFINIÇÃO DO PROBLEMA	7
1.2	JUSTIFICATIVA	7
1.3	OBJETIVO GERAL	7
1.4	OBJETIVOS ESPECÍFICOS	8
2	FUNDAMENTAÇÃO TEÓRICA	9
2.1	PROCESSAMENTO DIGITAL DE IMAGENS	9
2.1.1	Espectro Eletromagnético e Visão Computacional	10
2.2	RECONHECIMENTO BIOMÉTRICO	11
2.3	RECONHECIMENTO POR PADRONIZAÇÃO VASCULAR	14
3	DESENVOLVIMENTO	17
3.1	PRÉ PROCESSAMENTO	18
3.2	FILTROS E ELIMINAÇÃO DE RUÍDOS	20
3.3	BINARIZAÇÃO	22
3.4	OPERAÇÕES MORFOLÓGICAS	23
3.4.1	Erosão	23
3.4.2	Dilatação	24
3.4.3	Abertura	24
3.4.4	Fechamento	25
3.4.5	Máscara de Filtro	26
3.4.6	Histograma	26
3.5	SEGMENTAÇÃO E REGIÃO DE INTERESSE	27
3.6	DETECÇÃO DE BORDAS	27
4	CLASSIFICADORES E APRENDIZADO DE MÁQUINA	31
4.1	GRADIENTE DA IMAGEM	31
4.2	MÁQUINA DE VETORES DE SUPORTE - SVM	33

5	RESULTADOS E DISCUSSÕES	35
6	CONCLUSÃO	38
	REFERÊNCIAS	39

1 INTRODUÇÃO

O reconhecimento de indivíduos é natural para os seres humanos, inconscientemente, ao longo de toda a vida, padrões são identificados e reconhecidos pelo cérebro, criando uma biblioteca de rostos na mente. A aquisição de imagens é feita algumas poucas vezes, até que a imagem se torne familiar, e então o processamento necessário para o reconhecimento vai se tornando menor, o cérebro precisa buscar apenas entre as imagens previamente gravadas, a qual indivíduo aquele rosto pertence. Sistemas de inteligência artificial focados em visão computacional praticam o reconhecimento imitando o comportamento dos olhos humanos, para aquisição, e do cérebro, para processamento (FILHO O. M.; NETO, 1999) (GONZALEZ, 2000).

Biometria é a tecnologia utilizada para identificar usuários, baseando-se em características comportamentais ou fisiológicas de cada indivíduo. Dentre as medições biométricas se encontram a digital, face, íris e reconhecimento por voz. O reconhecimento facial por meio de processamento digital têm evoluído ano a ano, ganhando importância em diferentes setores de pesquisa, como aplicações de segurança pública ou privada, controle de tráfego, cadastramento, entre outros. A evolução de sistemas de reconhecimento de indivíduos é constante, e com isso a meta é a busca por resultados mais exatos e significativos. O reconhecimento facial possui muitas vantagens quando comparado a outros meios de identificação pessoal. O simples fato do rosto estar geralmente exposto facilita a aquisição de dados, e com isso auxiliando no treinamento e aprendizado de algoritmos de reconhecimento. O mapeamento de veias é considerado um dos métodos mais seguros e confiáveis de reconhecimento pessoal, pois os caminhos percorridos pelas veias não sofrem alterações ao longo dos anos, como é o caso da impressão digital que se desgasta, esses caminhos também independem de muitos fatores para coleta de dados, sendo necessário apenas um bom dispositivo de emissão de luz infravermelho, é um método de precisão, além de utilizar uma informação genética impossível de fraudar (MISHRA KANDERP NARAYAN MISHRA, 2016) (WHAT-WHEN-HOW, 2016).

A maior parte dos trabalhos de reconhecimento facial em processamento digital de imagem utilizam imagens na faixa visível ao olho humano do espectro eletromagnético. Entretanto, problemas como fotos tiradas em locais com baixa iluminação ou

sombreamento, são fatores de grande impacto à essas imagens, pois acabam prejudicando a extração de dados essenciais para o projeto e afetando o funcionamento dos algoritmos de reconhecimento. Estudos mais recentes comprovam que o uso de imagens térmicas, também conhecidas como imagens infra-vermelho, para esse propósito podem resolver falhas de processamento recorrentes em reconhecimento facial trabalhado sobre imagens do espectro visível, já que não sofrem alterações por fatores como luminosidade (BIOINFO, 2012) (BOURLAI ARUN ROSS, 2017).

1.1 DEFINIÇÃO DO PROBLEMA

O avanço dos sistemas digitais trouxe consigo uma preocupação gradativa com a segurança em sistemas de identificação pessoal. Com o avanço da tecnologia, métodos utilizados por muitos anos, como impressão digital, e identificação da íris tornam-se cada vez mais propícios à fraude, e com isso algoritmos para identificação de indivíduos precisam de mais precisão, velocidade de processamento e confiabilidade.

1.2 JUSTIFICATIVA

Buscando garantir a identificação correta dos indivíduos cadastrados, o mapeamento de veias têm se provado o método mais seguro de reconhecimento, tendo em vista que o desenho das veias não se altera com o passar dos anos e é impossível de ser clonado. Porém, esse método ainda não foi adotado como principal em análises forenses.

Existem atualmente diversos estudos e aplicações para mapeamento de veias dos dedos e mãos. Entretanto, no âmbito de mapeamento de veias faciais para reconhecimento de indivíduos o desenvolvimento está embrionário, possibilitando novas propostas de projetos de segurança tecnológica. Dessa forma, o trabalho aqui proposto consiste em buscar uma forma de padronizar imagens e caracterizar indivíduos, a fim de obter uma resposta satisfatória de resultados, auxiliando na comprovação da individualidade do mapeamento de veias.

1.3 OBJETIVO GERAL

Fazendo uso de um banco de imagens da Universidade de Notre Dame, que inclui imagens de mais de 50 indivíduos, em diferentes ângulos, luminosidades e

distâncias, o foco deste trabalho se dá pela caracterização e classificação de padrões para reconhecimento de face utilizando as características individuais de mapeamento de veias através do desenvolvimento de um algoritmo em linguagem Python, para extração das veias - características, seguido de um algoritmo classificador implementado em Matlab para reconhecimento de indivíduos.

1.4 OBJETIVOS ESPECÍFICOS

- Obtenção de um banco de imagens térmicas específico para processamento e obtenção de veias
- Desenvolvimento de algoritmo em Python para processamento de imagens, extração de características e armazenamento de dados de interesse
- Segmentação e extração de características utilizando informações extraídas da imagem por gradientes orientados *HOG*
- Classificação dos dados em padrões de indivíduos, utilizando técnica de aprendizado de máquina
- Teste e validação do algoritmo classificador de vetores de suporte *SVM*
- Comparação de resultados obtidos neste estudo com trabalhos anteriores de mapeamento de veias

2 FUNDAMENTAÇÃO TEÓRICA

O trabalho aqui proposto irá utilizar conhecimentos prévios em processamento de imagens e classificadores, juntamente com a análise do estado-da-arte em reconhecimento de indivíduos por parametrização biométrica.

2.1 PROCESSAMENTO DIGITAL DE IMAGENS

A partir dos anos 60 e 70, o uso de processamento digital de imagens (PDI) é cada vez mais expressivo, em diversas áreas e com inúmeras aplicações. O uso de processamento de imagens para diagnósticos médicos, vide Figura 2.1 e ressonância magnética são exemplos dentre diversos outros na área da medicina. Outra área com aumento expressivo do uso de processamento de imagens para análise de imagens microscópicas, por exemplo, é a biologia, com isso faz-se aumento de precisão, o que possibilita interpretação de informações captadas na imagem não visíveis a olho nu (FILHO O. M.; NETO, 1999).

FIGURA 2.1 – PDI na área médica



Fonte: Portal da Radiologia

Em áreas tecnológicas, o processamento digital de imagens é largamente aplicado, seja mapeando movimentação robótica com a utilização de uma câmeras, ou sensores de presença como sinais de entrada, como também aplicações em outras áreas da Engenharia, por exemplo em controle de tráfego. Cálculos avançados

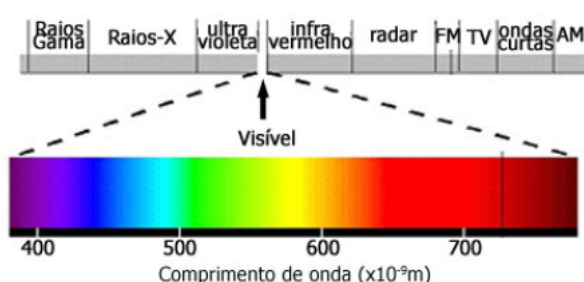
do processamento de imagens permitem que placas de veículos sejam reconhecidas utilizando vídeos capturados em rodovias ou em ruas movimentadas (CAMPOS SERGIO BAMPI, 2010).

Para que seja possível o processamento, as imagens digitais são representadas na forma de matrizes, onde cada elemento da matriz é equivalente a um pixel da imagem. Existem diversas informações atribuídas à uma imagem, que podem ser obtidas na análise pixel a pixel, como intensidade de cor e temperatura. Uma imagem colorida possui três informações de cor em cada ponto: luminância, matriz e saturação, que correspondem à luminosidade, comprimento de onda dominante e grau de pureza, respectivamente (PEDRINI, 2008).

2.1.1 Espectro Eletromagnético e Visão Computacional

A natureza da luz é explicada pelo estudo das ondas eletromagnéticas e pela constituição dos átomos. Sendo radiação eletromagnética, a luz apresenta um comportamento ondulatório caracterizado por sua frequência e comprimento de onda. Existe uma faixa do espectro eletromagnético, conforme pode ser visto na Figura 2.2, à qual o sistema visual humano é sensível, esta faixa se estende aproximadamente de 400 a 770 nm e denomina-se luz visível. Radiações eletromagnética com comprimentos de onda fora desta faixa não são percebidas pelo olho humano (BIOINFO, 2012).

FIGURA 2.2 – Espectro Eletromagnético



Fonte: bioinformaticacc

Objetos que emitem luz visível são percebidos em função da soma das cores espectrais emitidas pelos mesmos, através de processo aditivo. O processo aditivo pode ser interpretado como uma combinação variável em proporção de componentes monocromáticas nas faixas espectrais associadas às sensações de cor verde, vermelho

e azul, as quais são responsáveis pela formação de todas as cores no espectro visível. Assim, as cores verde, vermelho e azul são ditas cores primárias. Este processo de geração levou a concepção de um modelo cromático denominado RGB (*Red*, *Green*, e *Blue*). Nesse modelo, as cores são caracterizadas a partir de um código das três cores primárias, em uma escala que varia de 0 a 255. A combinação das três cores em seu valor mínimo (0 0 0), representa a cor preta, enquanto a combinação no valor máximo (255 255 255), representa a cor branca. A variação entre valores mínimos corresponde a tons escuros e entre os valores máximos, os tons são mais intensos, mais claros (BAKKER, 2004).

Comparando a visão computacional à forma de funcionamento dos olhos humanos, os olhos capturam uma imagem e o cérebro processa a imagem e trabalha com os adquiridos, os transformando em informações de memória. Na visão computacional, os olhos são o dispositivo de captura de imagem ou vídeo, uma câmera digital ou instrumento semelhante, o processamento da imagem é dividido em partes que simulam a forma como a mente trabalha uma informação, divide em informações menores a fim de obter características semelhantes à alguma memória do passado, e depois junta e interliga os resultados a fim de obter uma melhor compreensão do conjunto. A tabela 1 contém um comparativo entre a visão humana e a visão computacional, no que diz respeito à fatores de luminosidade e cor (BARROW, 1981).

2.2 RECONHECIMENTO BIOMÉTRICO

Um sistema biométrico é aquele que opera na aquisição de dados de indivíduos, extraindo características pertinentes ao sistema, e que serão usadas em reconhecimento de padrões, ao comparado com o banco de dados obtidos anteriormente, na fase de cadastramento (JAIN ARUN ROSS, 2004).

O termo biometria vem do grego, e seu significado é métrica da vida. De outra forma pode-se compreender como uma análise matemática das características individuais de cada pessoa, ou medidas de fatores biológicos únicas de cada indivíduo. Com informações biométricas, faz-se possível o cadastramento e o reconhecimento de indivíduos com uso de características próprias de cada indivíduo. Formas atuais de proteção de dados são senhas, chaves ou cartões, porém problemas atrelados à esses métodos são esquecimento, perda ou roubo desse *token* (SAVVIDES, 2010).

TABELA 1 – Comparação entre o sistema visual humano e um sistema de visão artificial

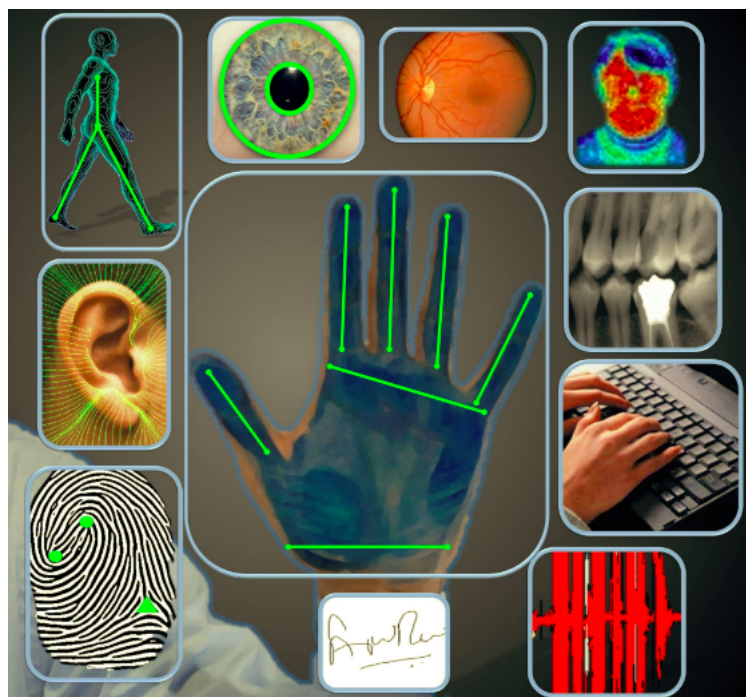
	Sistema visual humano	Sistema de visão artificial
Espectro	Limitado à faixa de luz visível (300 nm a 700 nm) do espectro de ondas eletromagnéticas.	Pode operar em praticamente todo o espectro de radiações eletromagnéticas, dos raios X ao infravermelho.
Flexibilidade	Extremamente flexível, capaz de se adaptar a diferentes tarefas e condições de trabalho.	Normalmente inflexível, apresenta bom desempenho somente na tarefa para a qual foi projetado.
Habilidade	Pode estabelecer estimativas relativamente precisas em assuntos subjetivos.	Pode efetuar medições exatas, baseadas em contagem de pixels e, portanto, dependentes da resolução da imagem digitalizada.
Cor	Possui capacidade de interpretação subjetiva de cores.	Mede objetivamente os valores das componentes R, G e B para determinação de cor.
Sensibilidade	Capaz de se adaptar a diferentes condições de luminosidade, características físicas da superfície do objeto e distância ao objeto. Limitado na distinção de muitos níveis diferentes de cinza, simultaneamente.	Sensível ao nível e padrão de iluminação, bem como à distância em relação ao objeto e suas características físicas. Pode trabalhar com centenas de tons de cinza, conforme projeto do digitalizador.
Tempo de resposta	Elevado, da ordem de 0,1 s.	Dependente de aspectos de hardware, podendo ser tão baixo quanto 0,001 s.
2-D e 3-D	Pode executar tarefas 3-D e com múltiplos comprimentos de onda (dentro do espectro de luz visível) facilmente.	Executa tarefas 2-D com relativa facilidade, mas é lento e limitado em tarefas 3-D.
Percepção	Percebe variações de brilho em escala logarítmica. A interpretação subjetiva de brilho depende da área ao redor do objeto considerado.	Pode perceber brilho em escala linear ou logarítmica.

Fonte: (FILHO O. M.; NETO, 1999)

Uma das maneiras primordiais de reconhecimento de indivíduos é a impressão digital, cada desenho é único, e por isso esse método foi utilizado por um longo período de tempo. Porém, com o passar dos anos, a impressão digital de um indivíduo sofre alterações, seja por desgaste, queimadura, doenças de pele ou cortes, fazendo com que esse método perca a precisão e fique apto à falhas e fraudes (OLIVEIRA, 2004)(JAIN ARUN ROSS, 2004).

Uma forma de desviar desses problemas, é utilizando o reconhecimento da íris,

FIGURA 2.3 – Sistemas biométricos



tendo em vista que a íris não sofre alterações ao longo da vida de um indivíduo. Instituições como seguradoras e planos de saúde utilizam desse meio para o reconhecimento pessoal e obtém um resultado satisfatório. Porém, em casos onde criminosos possuam conhecimento suficiente para aplicar engenharia reversa nos equipamentos de coleta e análise, o reconhecimento da íris torna-se suscetível à fraudes, com o uso de lentes especiais que enganam o sistema, e portanto não é mais considerado o método mais seguro (JAIN ARUN ROSS, 2004).

O primeiro sistema automatizado de reconhecimento facial foi desenvolvido por Takeo Kanade, em sua tese de doutorado em 1973. A partir daí surgiram trabalhos como a representação da face por *eigenfaces*. O método de *eigenfaces* consiste na comparação da face codificada com toda uma base de dados composta por faces codificadas de forma semelhante, é uma das formas mais intuitivas de classificar uma face. Este método utiliza uma maior quantidade de informação, para classificar as faces com base em padrões faciais genéricos, uma técnica semelhante à transformada de Fourier, utilizada muito frequentemente em aplicações que envolvem processamento de sinais, esta operação matemática consiste na decomposição de uma função principal em várias funções oscilatórias com parâmetros bem conhecidos, como por exemplo amplitude e frequência. Essa técnica foi desenvolvida com o objectivo de representar,

FIGURA 2.4 – Mapeamento de veias dos dedos



Fonte: Hitachi, Ltd.

de forma eficiente, imagens de faces através da análise de componentes principais (PCA - Principal Component Analysis), onde cada face pode ser representada como a combinação linear de diversas *eigenfaces*.

O nível de precisão dos sistemas atuais, em ambiente totalmente controlado (iluminação, posição do indivíduo e expressões faciais), o reconhecimento automatizado já ultrapassa o desempenho humano, principalmente quando a base de dados utilizada para treinamento e validação possui grande volume de faces. No entanto, ainda existem diversas preocupações quando se trata de um banco de imagens em ambiente não controlado (WHAT-WHEN-HOW, 2016) (LOPES, 2010).

2.3 RECONHECIMENTO POR PADRONIZAÇÃO VASCULAR

Padronização vascular é o nome dado ao método de reconhecimento não invasivo que baseia-se no padrão de veias de um indivíduo, seja nos dedos, na mão ou na face. Com a utilização de acessórios adaptados à geometria da mão, a leitura de dados pode se tornar muito mais rápida e exata. O processo de reconhecimento se dá pela comparação dos dados recém coletados, com os dados pré obtidos e tratados que estarão em um banco de dados próprio, até encontrar o padrão exato (HITACHI, 2012).

Sistemas de padronização vascular já são muito utilizados por agências bancárias e caixas eletrônicos em todo o país, por se tratar de um método onde as características independem do estado da epiderme, da idade, ou quaisquer outros

FIGURA 2.5 – Mapeamento de veias da mão

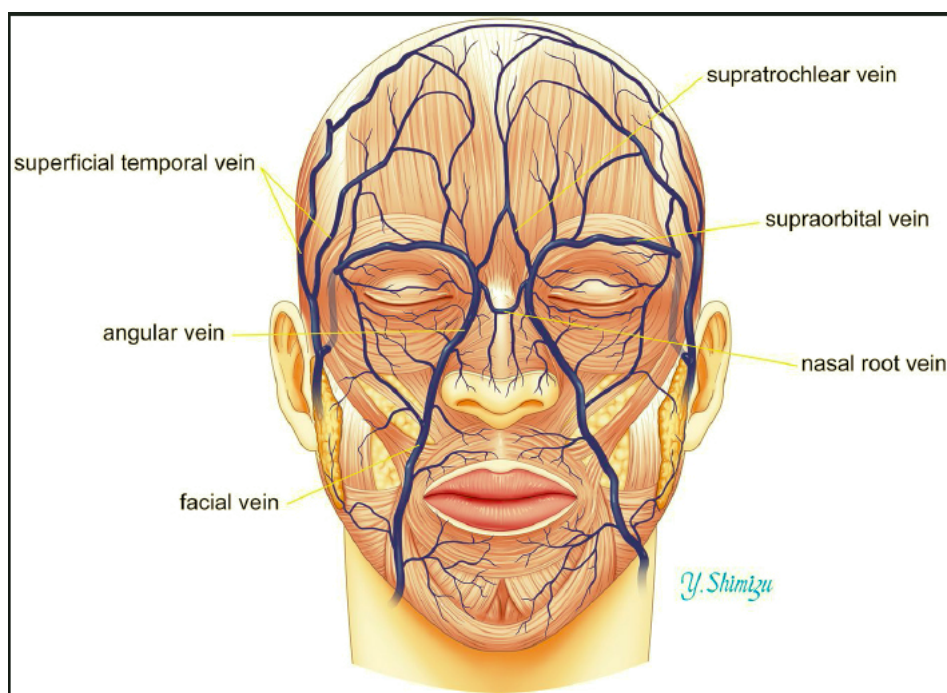


Fonte: NASA gov

fatores de grande influência em outros métodos de reconhecimento, garantindo assim maior confiabilidade para os usuários cadastrados. Analisando alguns desses métodos já implementados, é possível notar que o processo de cadastramento conta com uma coleta repetitiva de dados, o que contribui com a diminuição de erro quadrático resultante desses sistemas. Visando a padronização de veias, é necessário trabalhar com imagens no espectro eletromagnético infravermelho com comprimentos de onda entre $3\mu\text{m}$ e $14\mu\text{m}$. Câmeras especiais são capazes de capturar luzes infravermelho refletidas, detectando variações de temperatura na imagem (BOURLAI ARUN ROSS, 2017).

O funcionamento dos sistemas de mapeamento, é baseado na reação entre a luz infravermelho e o corpo humano. O dispositivo de reconhecimento emite raios de luz infravermelho em direção à pele humana, e à atravessa, ao entrar no tecido, as hemoglobinas do sangue então refletem a luz de forma que a câmera capture a imagem refletida. O processamento de imagens se inicia até que o padrão de veias seja definido. Como exemplo de sistemas de reconhecimento vascular, pode-se citar o reconhecimento pelas veias dos dedos, como mostrado na Figura 2.4, um método comumente usado para cadastramento de título eleitoral e fechos digitais de portas. Já o método de reconhecimento de veias das mãos, mostrado na Figura 2.5, é utilizado no cadastramento de usuários em agências bancárias, e está implementado em caixas eletrônicos de forma à substituir as senhas, resultando em mais segurança para seus clientes (ABRÃO, 2008).

FIGURA 2.6 – Veias da face



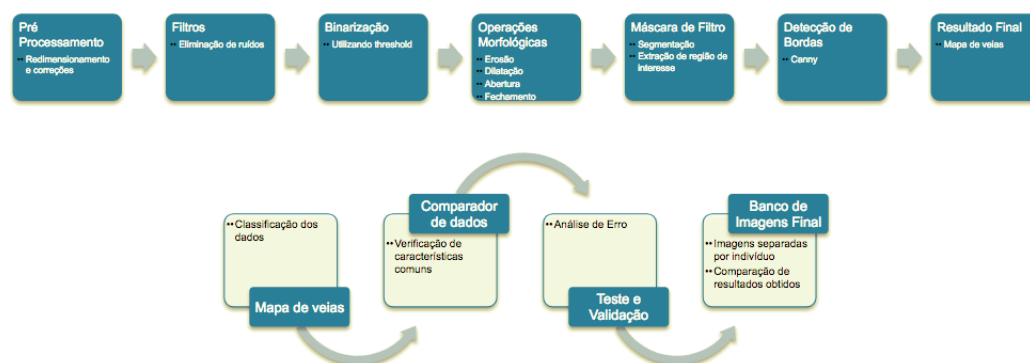
Fonte: ADAM Atlas de Anatomia

Para falar de veias da face, entretanto, não é fácil encontrar sistemas onde esse método já esteja implementado, diversos estudos buscam uma resposta satisfatória de precisão, utilizando técnicas de processamento, classificação e reconhecimento de forma rápida e de baixo custo. Para aquisição de imagens e vídeos em infravermelho, é necessário trabalhar com câmeras especiais que demandam maiores investimentos se comparadas a outros equipamentos técnicos para reconhecimento vascular. Porém em casos onde a segurança é prioridade, e há necessidade de um sistema mais eficaz de reconhecimento, o reconhecimento de veias faciais é indicado, pois seu processamento robusto conta com a garantia de que as veias da face são únicas e impossíveis de modificar, justificando o investimento no projeto. Na Figura 2.6 pode-se observar as principais veias que compõe a face, deve-se prestar atenção redobrada no momento de extração das características da imagem trabalhada, a fim de evitar falsos positivos em se tratando de pequenas veias, ou seja, considerando ruídos como pequenos vasos.

3 DESENVOLVIMENTO

As etapas que compõe um trabalho envolvendo processamento digital de imagem em reconhecimento biométrico consistem em: aquisição de imagens usando um aparelho de captura, no caso desse trabalho uma câmera térmica especial; do pré-processamento, preparação da imagem a ser trabalhada; melhorias na luminosidade e posicionamento das imagens; eliminação de ruídos e possíveis distorções presentes na imagem com aplicação de filtros; extração das regiões de interesse, especificamente a região da face; caracterização das informações extraídas, a fim de diminuir o processamento na fase de classificação; classificação de imagens a partir de informações comuns, utilizando algoritmo inteligente de seleção. O diagrama da Figura 3.1 mostra o caminho que será percorrido durante o desenvolvimento desse projeto. Sendo a parte superior do diagrama as atividades de processamento digital de imagens, que compõem os primeiros seis meses de desenvolvimento. E a parte inferior do diagrama mostra as atividades de *machine learning* e classificação dos dados que serão executadas no segundo semestre de desenvolvimento.

FIGURA 3.1 – Diagrama de Atividades

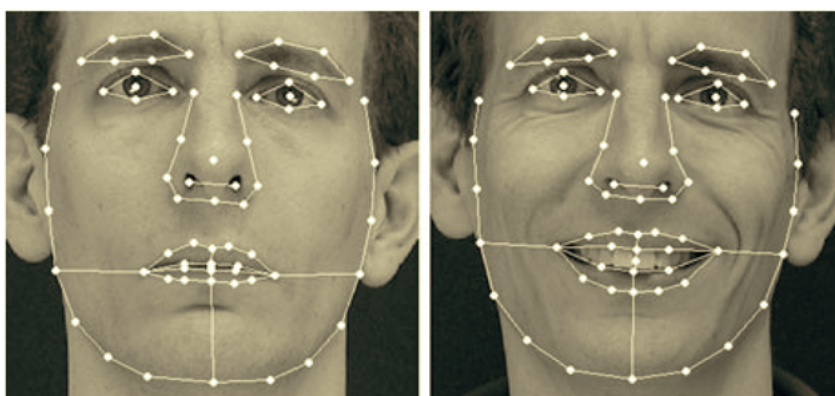


Fonte: Autora

Os primeiros passos para esse trabalho se deram por obter acesso à um vasto banco de imagens infravermelho Collection X1, concedidas pela Universidade de Notre Dame. Para obtenção das 2294 imagens contidas no banco, uma câmera LWIR (Long Wave Infrared) foi utilizada. As imagens não estão normalizadas, isto é foram obtidas em diferentes cenários, ângulos e condições climáticas.

A fim de obter resultados comparativos à estudos anteriores processados em Matlab, o desenvolvimento desse trabalho será feito utilizando linguagem Python com uso da biblioteca própria de processamento de imagem e visão computacional OpenCV, disponível também para programação em C ou Java, e bibliotecas de métodos matemáticos NumPy, SciPy e Matplotlib. Após análise e embasamento teórico sobre tratamento de imagens infravermelho, foi possível dar início ao desenvolvimento seguindo os seguintes passos.

FIGURA 3.2 – Modelagem de Face



Fonte: What-When-How

3.1 PRÉ PROCESSAMENTO

Essa etapa consiste em pré processar as imagens de forma automatizada, de forma a obter um resultado geral mais apropriado para o processamento. Como as imagens visíveis do banco possuíam dimensões de 1200x1600, largas dimensões e com isso eram muito pesadas, o processamento poderia se tornar lento. Com isso, um primeiro script de redimensionamento de imagens foi criado, para obter um tamanho que não reduzisse o desempenho do código de reconhecimento, mas também não carregasse o processamento afetando o tempo de resposta do programa. O script foi criado em linguagem Python, e seu funcionamento era procurar as imagens térmicas '.tiff', calcular suas medidas de altura e comprimento, e redimensionar as imagens visíveis '.jpg' de acordo com essas escalas obtidas, sem distorções de imagem, em seguida gravar as imagens em uma nova pasta que será usada ao longo desse projeto. Parte do código criado é mostrado na Figura 3.3, onde a coleta de informações sobre a imagem infravermelho é repassada para a imagem visível. Por se tratar de um

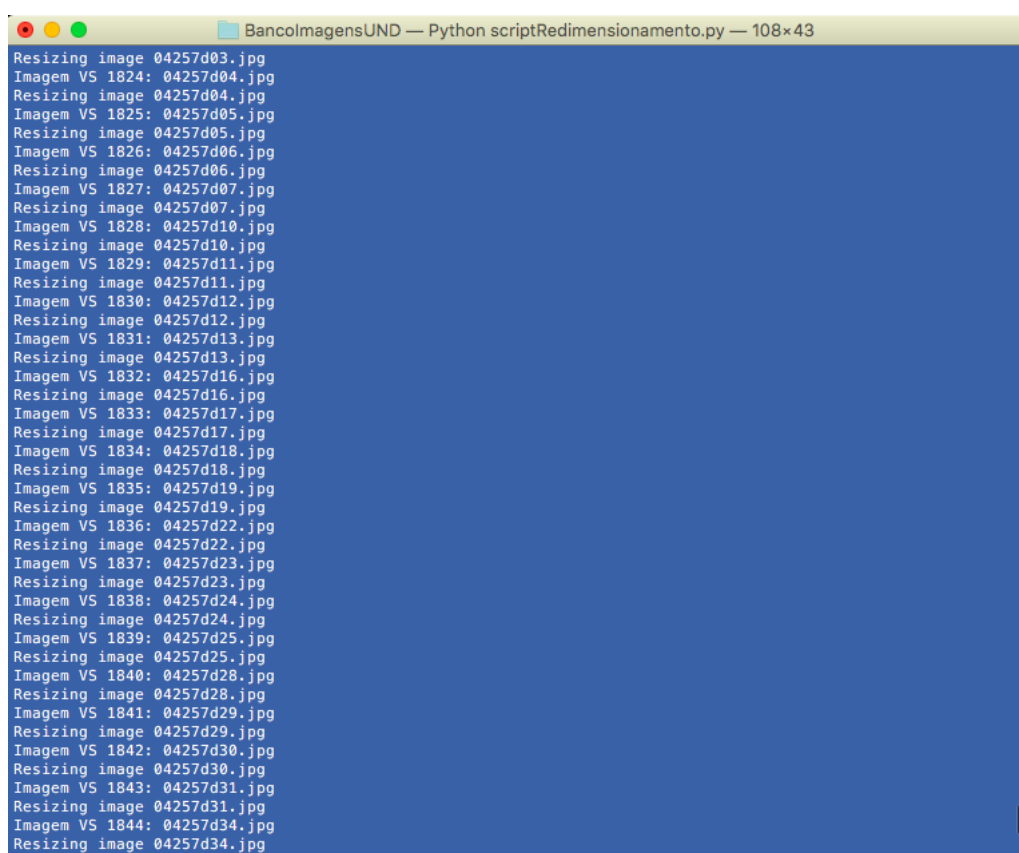
banco com 2290 imagens, o redimensionamento manual demandaria certo tempo e esforço que foram poupados, e o resultado obtido foi o esperado, satisfazendo a economia de memória de processamento. O script leva cerca de 3 minutos para redimensionar todas as imagens, a resposta do compilador pode ser vista na Figura 3.4. As imagens visíveis resultantes, Figura 3.5, possuem normalização de distâncias com as imagens infravermelhas, possibilitando a utilização dessas para obtenção de parâmetros importantes, como distância euclidiana entre olhos, por exemplo.

FIGURA 3.3 – Código: Código de redimensionamento contido no script

```
imgVS      = cv2.imread(filesVS[a],cv2.IMREAD_UNCHANGED)
height, width = imgIR.shape[:2]
resVS = cv2.resize(imgVS,(width, height), interpolation = cv2.INTER_CUBIC)
cv2.imwrite(str(fileNameVS),resVS)
```

Fonte: Autora

FIGURA 3.4 – Terminal - Resposta do compilador ao script de redimensionamento



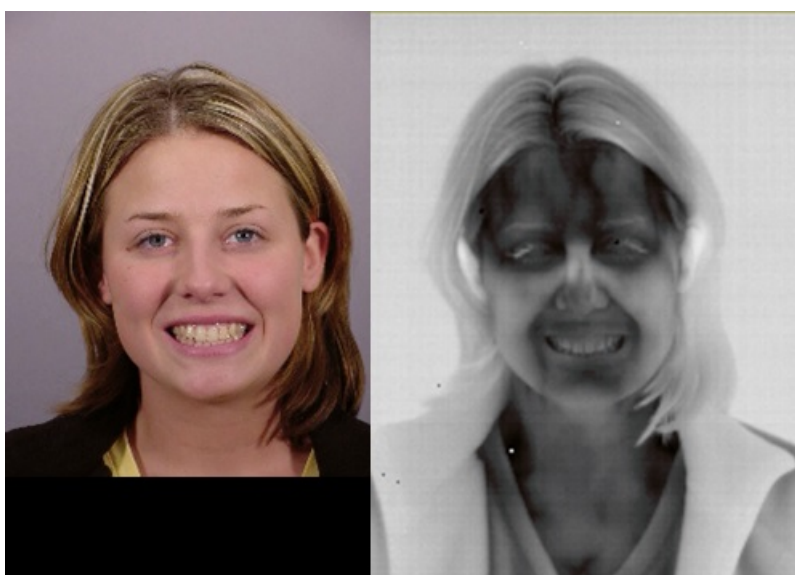
```
BancoImagensUND — Python scriptRedimensionamento.py — 108x43
Resizing image 04257d03.jpg
Imagem VS 1824: 04257d04.jpg
Resizing image 04257d04.jpg
Imagem VS 1825: 04257d05.jpg
Resizing image 04257d05.jpg
Imagem VS 1826: 04257d06.jpg
Resizing image 04257d06.jpg
Imagem VS 1827: 04257d07.jpg
Resizing image 04257d07.jpg
Imagem VS 1828: 04257d10.jpg
Resizing image 04257d10.jpg
Imagem VS 1829: 04257d11.jpg
Resizing image 04257d11.jpg
Imagem VS 1830: 04257d12.jpg
Resizing image 04257d12.jpg
Imagem VS 1831: 04257d13.jpg
Resizing image 04257d13.jpg
Imagem VS 1832: 04257d16.jpg
Resizing image 04257d16.jpg
Imagem VS 1833: 04257d17.jpg
Resizing image 04257d17.jpg
Imagem VS 1834: 04257d18.jpg
Resizing image 04257d18.jpg
Imagem VS 1835: 04257d19.jpg
Resizing image 04257d19.jpg
Imagem VS 1836: 04257d22.jpg
Resizing image 04257d22.jpg
Imagem VS 1837: 04257d23.jpg
Resizing image 04257d23.jpg
Imagem VS 1838: 04257d24.jpg
Resizing image 04257d24.jpg
Imagem VS 1839: 04257d25.jpg
Resizing image 04257d25.jpg
Imagem VS 1840: 04257d28.jpg
Resizing image 04257d28.jpg
Imagem VS 1841: 04257d29.jpg
Resizing image 04257d29.jpg
Imagem VS 1842: 04257d30.jpg
Resizing image 04257d30.jpg
Imagem VS 1843: 04257d31.jpg
Resizing image 04257d31.jpg
Imagem VS 1844: 04257d34.jpg
Resizing image 04257d34.jpg
```

Fonte: Autora

Ainda no script de pré-processamento, ajustes seguintes foram aplicados.

Esses ajustes de imagem, como realce, brilho e correção de ângulo, foram aplicados com objetivo de normalizar as imagens para futuro processamento e melhor extração de dados. As imagens térmicas posteriormente serão comparadas com as imagens reais para validação de resultados. O principal objetivo das técnicas de realce de imagem é o de melhorar aspectos a fim de torná-las mais apropriadas para a dada aplicação, nesse caso, algumas imagens continham baixa luminosidade, ou pouco contraste entre a pessoa e o fundo, e essas técnicas de realce foram importantes para melhorar o desempenho do projeto. Poucas técnicas de angulação foram necessárias nesse primeiro momento, pois as fotos foram tiradas com indivíduos sentados, com diferentes distanciamentos da câmera.

FIGURA 3.5 – Imagem RGB e térmica após script de normalização

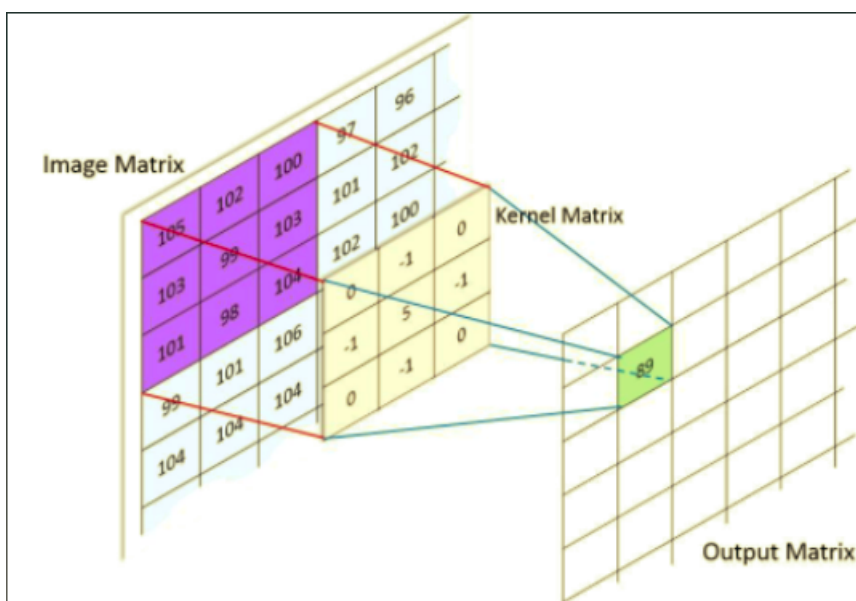


Fonte: Autora

3.2 FILTROS E ELIMINAÇÃO DE RUÍDOS

Após o pré processamento, inicia-se a filtragem. Para análise pixel a pixel, é importante compreender o que significa cada informação da imagem. A conectividade de pixels mostra onde estão os objetos da imagem, dessa forma, pixels com baixa conectividade são considerados ruídos. A conectividade pode ser analisada por uma vizinhança de 4 ou 8 pixels, de forma a verificar se aquele pixel em questão faz parte de um mesmo objeto, de um novo objeto, ou se trata de ruído (GONZALEZ, 2000).

FIGURA 3.6 – Convolução entre imagem original e kernel



Fonte: (GURU, 2017)

Filtros digitais são utilizados para separação de sinais em diferentes frequências, ou para restauração de dados. Técnicas de filtragem podem ser feitas no domínio espacial ou no domínio da frequência utilizando operações de convolução. Convolução é um dos procedimentos mais importantes em processamento de sinais ou imagens, e pode operar em sinais de uma dimensão, ou 1D, que são processamentos em sinais de áudio, 2D que são processamentos em imagens e 3D que são processamentos em vídeos. Cada operação de convolução precisa de um kernel, que é uma matriz menor que aquela que compõe a imagem original em altura e largura. A criação do kernel dependerá do resultado desejado na operação da imagem, e ele pode implicar em aumento de nitidez, ofuscação, detecção de borda, entre outros.

FIGURA 3.7 – Filtragem por Gradiente e Filtragem por Médias



Fonte: Autora

FIGURA 3.8 – Filtragem por Gradiente e Filtragem por Médias



Fonte: Autora

FIGURA 3.9 – Filtragem por Gradiente e Filtragem por Médias



Fonte: Autora

Para a filtragem inicial o objetivo era obter uma máxima separação entre o fundo da imagem, *background*, e o indivíduo, *foreground*. Testes foram feitos utilizando filtros de média e de gradiente, filtros foram aplicados à imagem pré-tratada. Observando os resultados obtidos nas Figuras 3.7, 3.8 e 3.9, é possível observar que o filtro de médias aplicado à imagem resultou na integralização do ruído de fundo, criando blocos na imagem, de forma a facilitar a remoção de dados insignificantes ao processo.

3.3 BINARIZAÇÃO

O realce da imagem por *threshold* é também conhecido como limiarização ou binarização. Consiste em transformar todos os pixels de uma imagem de modo que eles tenham apenas 2 valores, preto e branco (0 e 1). O negativo binário de uma imagem é obtido a partir de operação de negação aos pixels da imagem $f(x,y)$. Assim, todos os valores de pixel da imagem original são invertidos logicamente, aqueles pretos ficam brancos, e os brancos ficam pretos.

Após passar pelo procedimento de filtragem, é possível calcular um valor de *threshold* limite entre o indivíduo e o fundo, identificando assim um ponto onde valores abaixo deste sejam considerados descartáveis, e valores acima sejam considerados de interesse. A fim de encontrar uma melhor resposta para diferentes ambientes e índices luminosos, o teste de binarização foi feito também com valores negativos de binaridade, para o caso onde os dados descartáveis possuam valores mais altos.

Dessa forma, a imagem binarizada do indivíduo citado na Figura 3.7 são mostrados na Figura 3.10.

FIGURA 3.10 – Imagem Binária Resultante



Fonte: Autora

3.4 OPERAÇÕES MORFOLÓGICAS

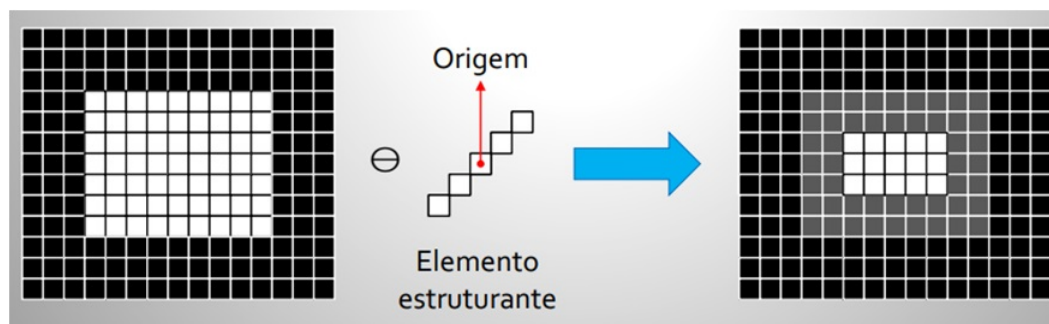
A partir daqui, com a obtenção de uma imagem binarizada, métodos computacionais de processamento podem ser aplicados, pois os pixels agora estão transformados em bits, possibilitando operações lógicas à imagem. A morfologia matemática pode ser aplicada em diversas áreas de processamento e análise de imagens, com objetivos tão distintos como realce, filtragem, segmentação, detecção de bordas, esqueletização, afinamento, dentre outras (FILHO O. M.; NETO, 1999). As operações morfológicas utilizadas no projeto em questão, serão explicadas mais detalhadamente.

3.4.1 Erosão

Erosão faz uma análise de conjuntos entre o *kernel* e a imagem, resultando nos bits do *kernel* que estão contidos na imagem. Seu resultado é uma afinação de

bits e depende do *kernel* escolhido pelo desenvolvedor.

FIGURA 3.11 – Operação de erosão

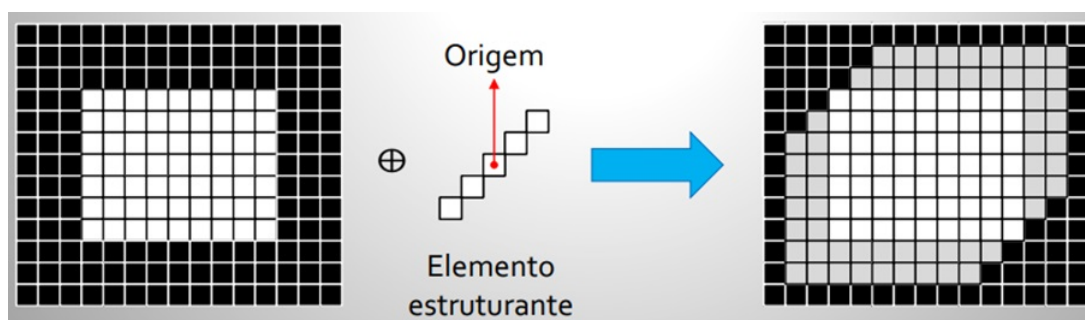


Fonte: EESC-USP

3.4.2 Dilatação

Assim como a erosão, dilatação trata-se de uma manipulação de conjuntos, fazendo com a interseção entre a imagem e o *kernel*, e sua distância relativa sejam o resultado da operação de dilatação de bits. O resultado da dilatação é uma expansão controlada de bits, que depende do *kernel* escolhido pelo desenvolvedor.

FIGURA 3.12 – Operação de dilatação

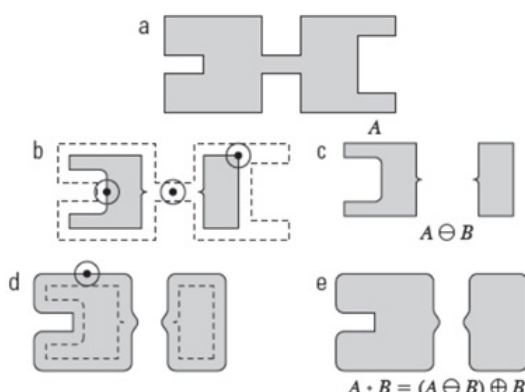


Fonte: EESC-USP

3.4.3 Abertura

A abertura é uma operação que suaviza contornos em uma imagem binária, de forma que pixels ligados pobremente serão eliminados. Isso quer dizer que ruídos restantes das operações anteriores na imagem serão reduzidos e eliminados.

FIGURA 3.13 – Operação de abertura



Fonte: EESC-USP

Simplificando em termos, a abertura é uma erosão seguida de dilatação. Em Python com a biblioteca OpenCV, a abertura se dá pela função:

```
opening = cv2.morphologyEx(BW,cv2.MORPHOPEN,kernel, iterations)
```

Onde BW é uma imagem binária, e *kernel* é a matriz *kernel* criada para a operação, e iterations é o número de iterações a ser aplicado. No projeto em questão o *kernel* utilizado foi uma matriz de 1's 5x5:

```
kernel = np.ones((5,5),np.uint8)
```

O resultado da operação de abertura é a remoção de objetos que estejam conectados por menos bits que o tamanho do *kernel*. Nesse projeto a operação de abertura foi realizada em 6 iterações.

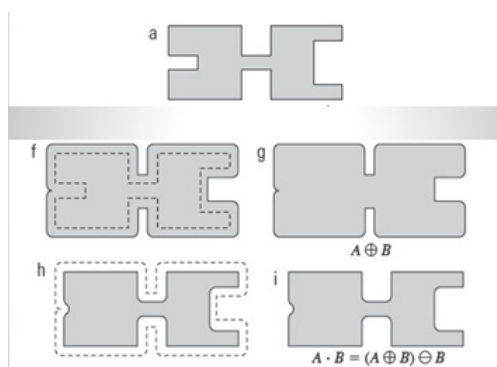
3.4.4 Fechamento

O fechamento é uma operação que acentua contornos em uma imagem binária, de forma que pixels separados pobremente serão unidos. Isso quer dizer que se um objeto foi separado erroneamente, seus bits serão unidos, de forma à preencher falhas na máscara da imagem.

Novamente simplificando em termos, o fechamento é uma dilatação seguida de erosão. Em Python com a biblioteca OpenCV, o fechamento se dá pela função:

```
closure = cv2.morphologyEx(IM,cv2.MORPHCLOSE,kernel, iterations)
```

FIGURA 3.14 – Operação de fechamento



Fonte: EESC-USP

onde IM é uma imagem binária ou em escala de cinza, e *kernel* é a matriz *kernel* criada para a operação. Neste projeto, a operação morfológica de fechamento foi aplicada à imagem binária resultante da abertura, utilizando o mesmo *kernel* com 4 iterações a fim de recuperar dados perdidos.

3.4.5 Máscara de Filtro

A máscara é a imagem binarizada resultante após aplicação dos procedimentos de extração de objetos. Operações morfológicas resultam em diferentes imagens, que podem ser trabalhadas de diferentes formas dependendo do objetivo final do desenvolvedor. Para o caso de obter como resultado final, unicamente dados da região da face, subtração da imagem original e imagens foram

3.4.6 Histograma

Após obter a imagem em tons de cinza, é possível relacionar o número de pixels correspondentes a um certo tom de cinza e as probabilidades de nível de cinza na imagem. Essa relação pode ser mostrada graficamente, na forma de histograma. Um histograma apresenta várias características importantes da imagem, como a probabilidade de um pixel da imagem apresentar nível de cinza no tom indicado. Portanto, um histograma é uma função de distribuição de probabilidades e como tal deve obedecer aos axiomas e teoremas da teoria de probabilidade, como a soma das probabilidades sempre será igual a 1. O histograma é utilizado para identificar grandes variações de informação da imagem, a fim de obter melhores resultados de extração de região de interesse (GONZALEZ, 2000).

3.5 SEGMENTAÇÃO E REGIÃO DE INTERESSE

O processo seguinte é subtração de dados da imagem original à máscara binária, a primeira segmentação. Obtendo como resultado uma imagem com menor quantidade de informação, extraindo apenas o indivíduo e eliminando todos os dados que não são interessantes para a análise.

Com a imagem resultante, filtrada pelo filtro de médias, o passo seguinte se deu por criar um *kernel* de bits 1, e tamanho 5x5. Esse foi utilizado para operações morfológicas na imagem, de forma a passar por abertura em 6 iterações, fechamento em 4 iterações e erosão por outras 6 iterações, o resultado obtido pode ser visto nas Figuras 3.15, 3.16 e 3.17. Nessas imagens resultantes, nota-se que a extração está satisfatória, pois dados da região da face não foram perdidos durante o processo, mas ainda será necessário realizar outras etapas de segmentação, a fim de obter a menor região possível de dados, contendo apenas a região facial, como o resultado apresentado na Figura 3.16.

3.6 DETECÇÃO DE BORDAS

Algoritmos de detecção de bordas são aqueles que encontram as curvas unidas por pontos contínuos da imagem, formando suas terminações. Bordas possuem mesma coloração e intensidade de cor, dessa forma sua detecção é obtida através da análise de gradientes ponto a ponto, como variação brusca de cor, de direcionamento, ou descontinuidade de pixels.

Existem diversos métodos de detecção de bordas, para essa etapa do desenvolvimento foram testados os métodos de Sobel e Canny, sendo o de Canny aquele que obteve melhor resultado para extração de veias. Esse detector de bordas faz a primeira derivada de uma gaussiana, e aproxima o operador que otimiza o produto da razão sinal-ruído e sua localização.

Os objetivos da detecção de bordas de Canny segue três pontos chave para ser considerado ótimo: (a) Boa detecção - Baixa probabilidade de falhas nos pontos marcados como bordas, isso é detecções corretas de pontos. Esse critério se especifica à maximizar a razão sinal-ruído.

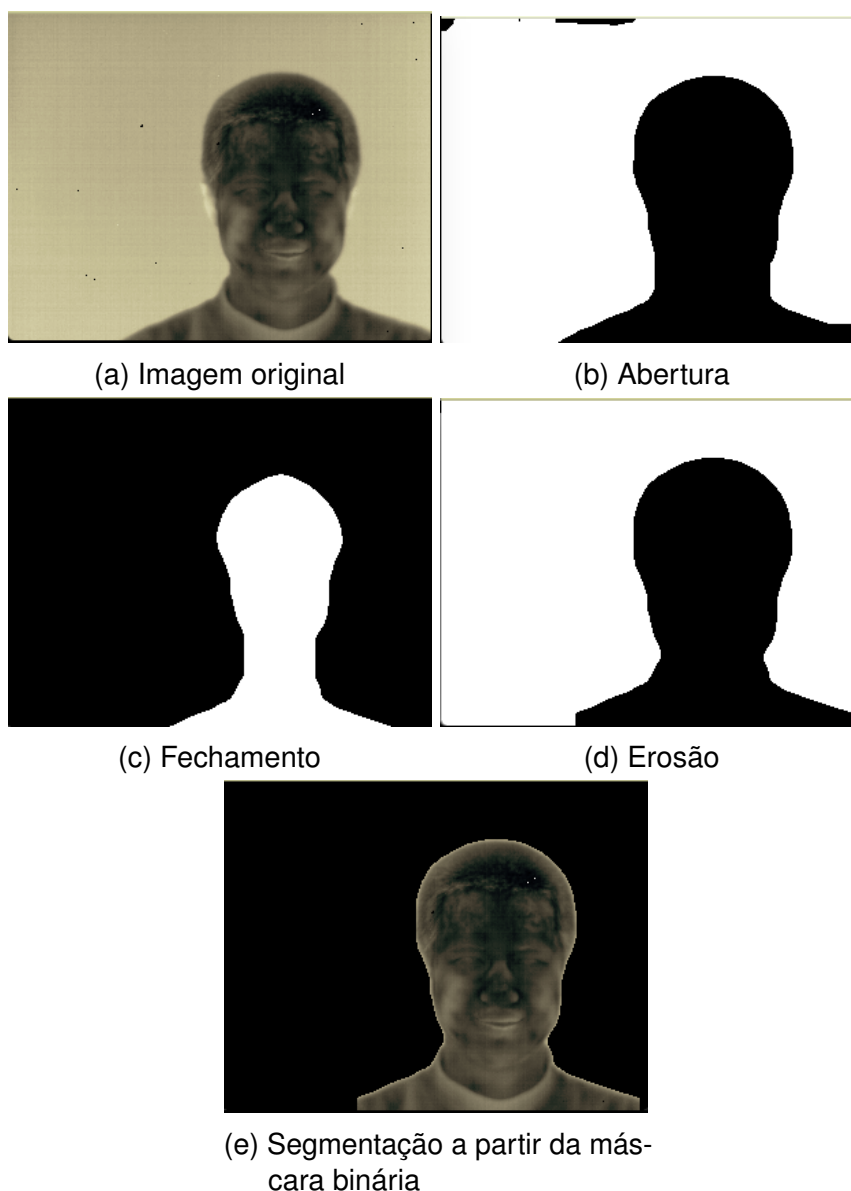


FIGURA 3.15 – Operações Morfológicas e Máscara Binária da Imagem

Fonte: Autora

(b) Boa localização - Os pontos marcados como bordas pelo operador precisam estar o mais próximo possível das bordas reais da imagem original.

(c) Resposta mínima - Apenas uma detecção por borda, sem repetir detecções.

Em Python, a utilização do método de Canny exige alguns parâmetros. Esse método possui dois valores de *threshold*, um superior e um inferior. Se o gradiente de pixel for maior que o *threshold* superior, o pixel é detectado como borda. Se o gradiente de pixel for menor que o *threshold* inferior, o pixel é rejeitado. Se o gradiente de pixel estiver entre os dois *threshold*, então esse será considerado borda caso esse pixel esteja conectado a outro que tenha valor maior que o *threshold* superior.

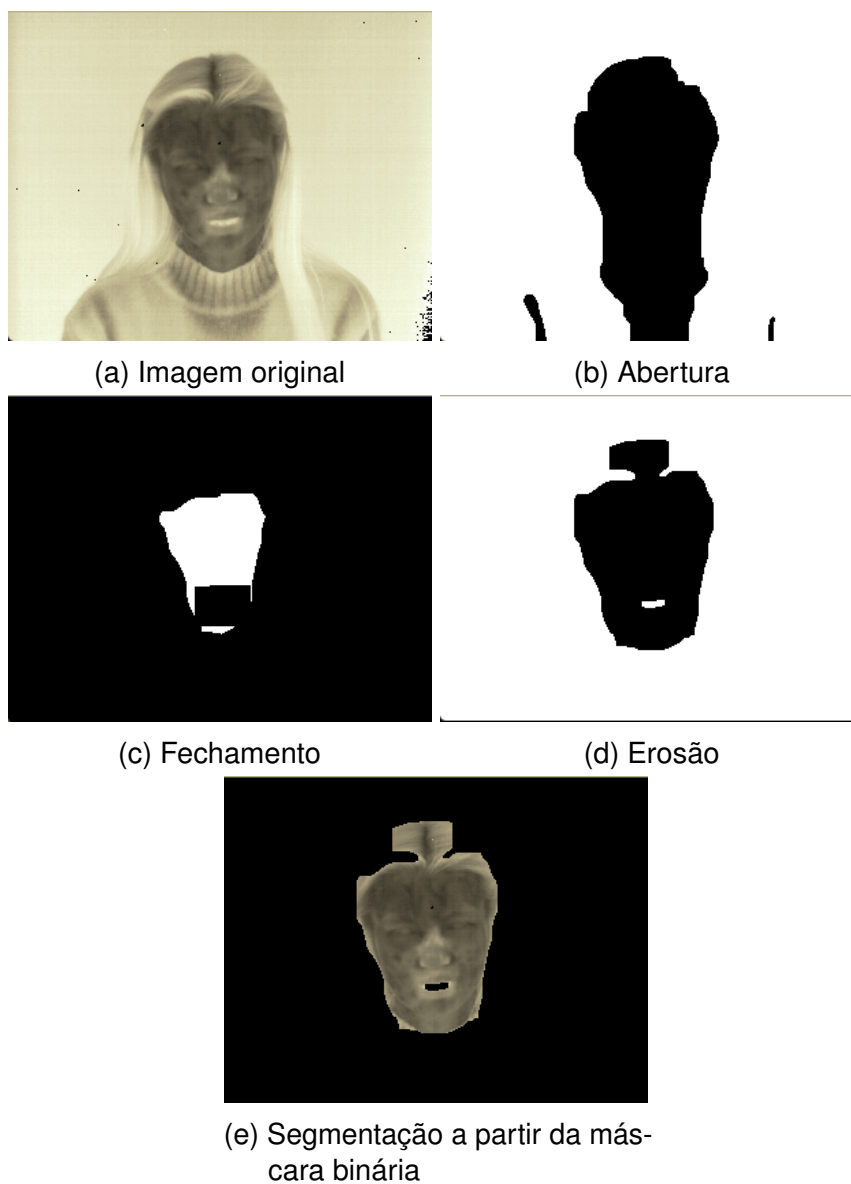


FIGURA 3.16 – Operações Morfológicas e Máscara Binária da Imagem

Fonte: Autora

Testes de detecção de borda para esse projeto foram realizados utilizando valores de *threshold* superior e inferior de:

- 50, 50
- 30, 60
- 60, 80

Sendo a resposta mais satisfatória obtida para valores de 60 e 80, onde eliminaram-se ruídos próximos as veias, restando vetores bem definidos que serão utilizados na etapa de classificação. O resultado pode ser observado na figura 3.18.

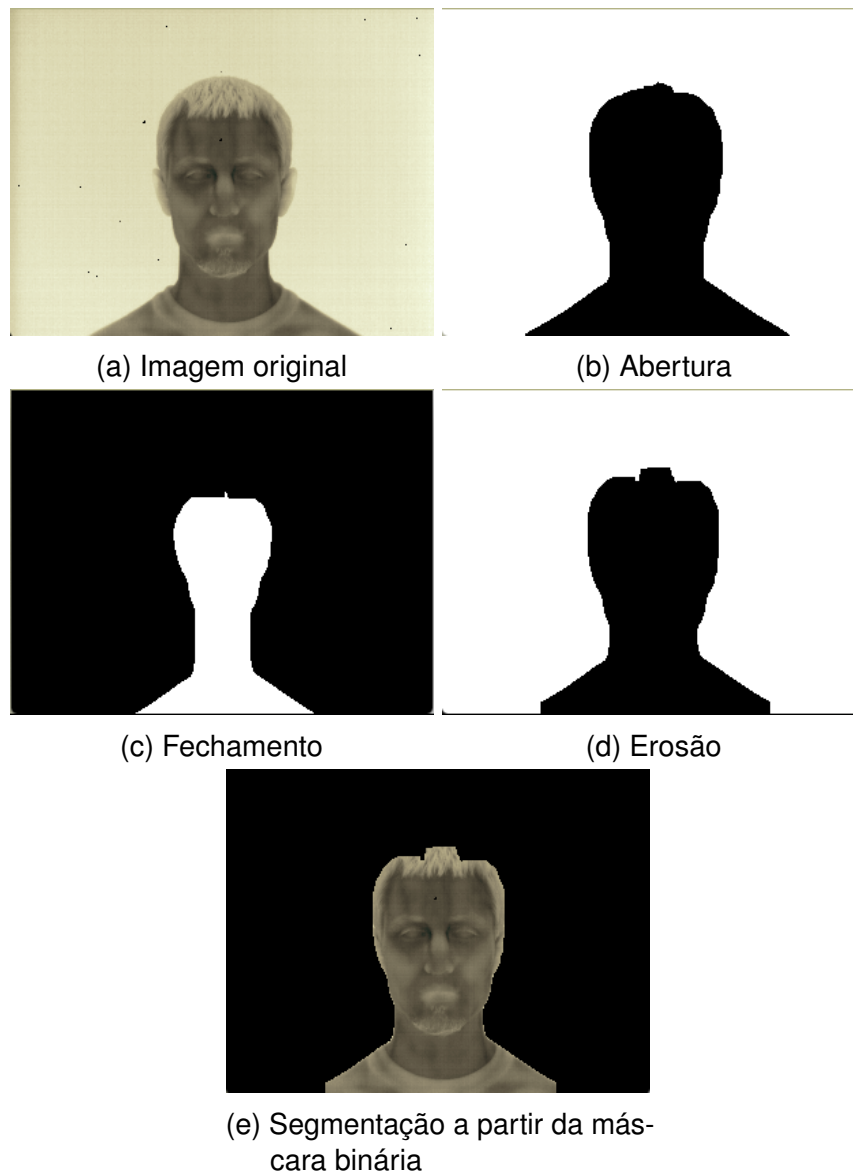


FIGURA 3.17 – Operações Morfológicas e Máscara Binária da Imagem

Fonte: Autora

FIGURA 3.18 – Gradiente Orientado



Fonte: Autora

4 CLASSIFICADORES E APRENDIZADO DE MÁQUINA

Ao falar em classificação por reconhecimento de padrões, os três conceitos básicos precisam ser compreendidos:

- Características: O vetor de características, que é definido por dados de informação extraídos na forma de medidas das instâncias. Essas medidas devem maximizar as semelhanças intraclasse e as diferenças extraclasse.
- Classe: Conjunto de instâncias que possuem características em comum.
- Classificação: Ato de rotular instâncias de acordo com suas características.

Dentre os métodos de classificação, existem o método supervisionado e o não supervisionado. As etapas da classificação supervisionada se dão pela escolha de um conjunto de treinamento, seguido da escolha de parâmetros relevantes a serem medidos, depois a obtenção da função discriminante - que pode ser obtida por método não estatístico (distribuição livre), ou estatístico - eliminação dos parâmetros não relevantes e por fim os testes, realizados com imagens fora das utilizadas no treinamento.

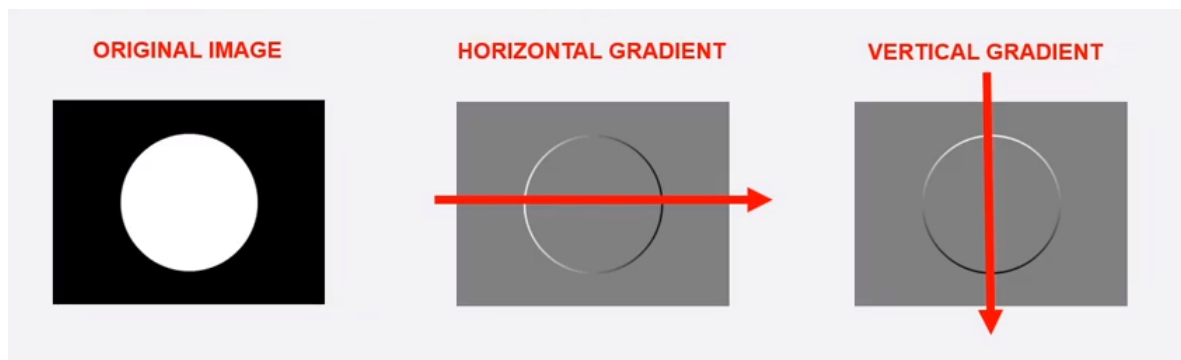
A partir dessa etapa, o projeto passa a ser desenvolvido em MATLAB, para que seja possível realizar a extração de características para o treinamento e a classificação das imagens. Foi necessário utilizar a biblioteca específica de aprendizado de máquina do programa MATLAB, *Computer Vision Toolbox*. A sequência de passos do trabalho será explicada a seguir, partindo da extração de características, para o treinamento do algoritmo e por fim a execução da classificação.

4.1 GRADIENTE DA IMAGEM

A extração de características da imagem pelo uso de gradientes orientados é uma técnica muito eficiente para detecção de objetos. A forma computacional de extração de dados é muito diferente da lógica humana, essa técnica utiliza medições matemáticas para criar padrões de comportamento, quando a imagem é analisada pixel a pixel.

O gradiente é a medida da modificação da imagem, em pixels de proximidade ou vizinhos, analisados na função. Para entender melhor como essa técnica funciona,

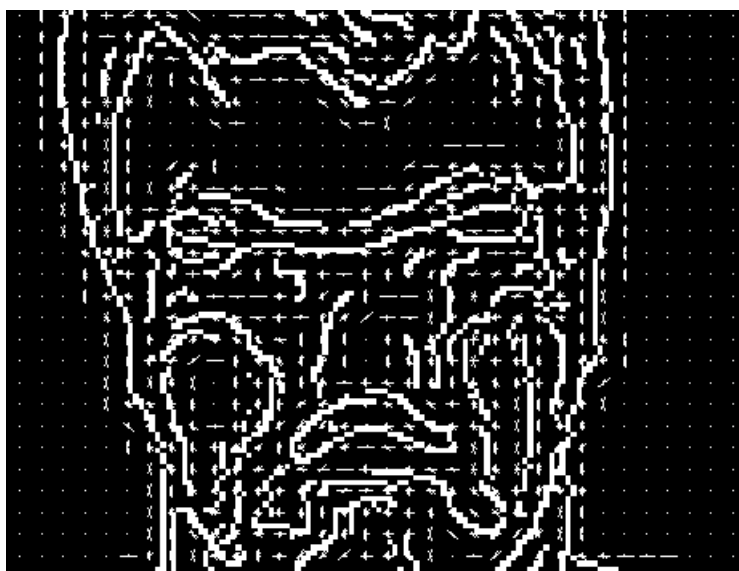
FIGURA 4.1 – Exemplo de Gradiente Orientado



Fonte: Udemy

matematicamente falando, a análise é feita pixel a pixel, e o gradiente da imagem é a derivada da função em x , e em y . Caso haja grande variação de intensidade, por exemplo vários pixels de fundo da imagem serão pretos, a partir do momento em que houver alteração para a cor branca, por exemplo, o gradiente indicará a intensidade da variação e a direção do vetor de mudança, como na figura 4.1, onde o valor dos pixels é de 0, até o momento em que inicia o círculo branco, como há aumento de intensidade, o gradiente será positivo e direcionado da esquerda para a direita.

FIGURA 4.2 – Gradiente Orientado



Fonte: Autora

Para o trabalho utilizado, a extração de características é realizada ao chamar a função *extractHOGFeatures* do MATLAB para cada imagem a ser analisada. O

FIGURA 4.3 – Gradiente Orientado - Aproximado



Fonte: Autora

resultado do gradiente em x e em y pode ser visto na figura 4.2, a figura 5.2 mostra a imagem aproximada, detalhando os gradientes encontrados.

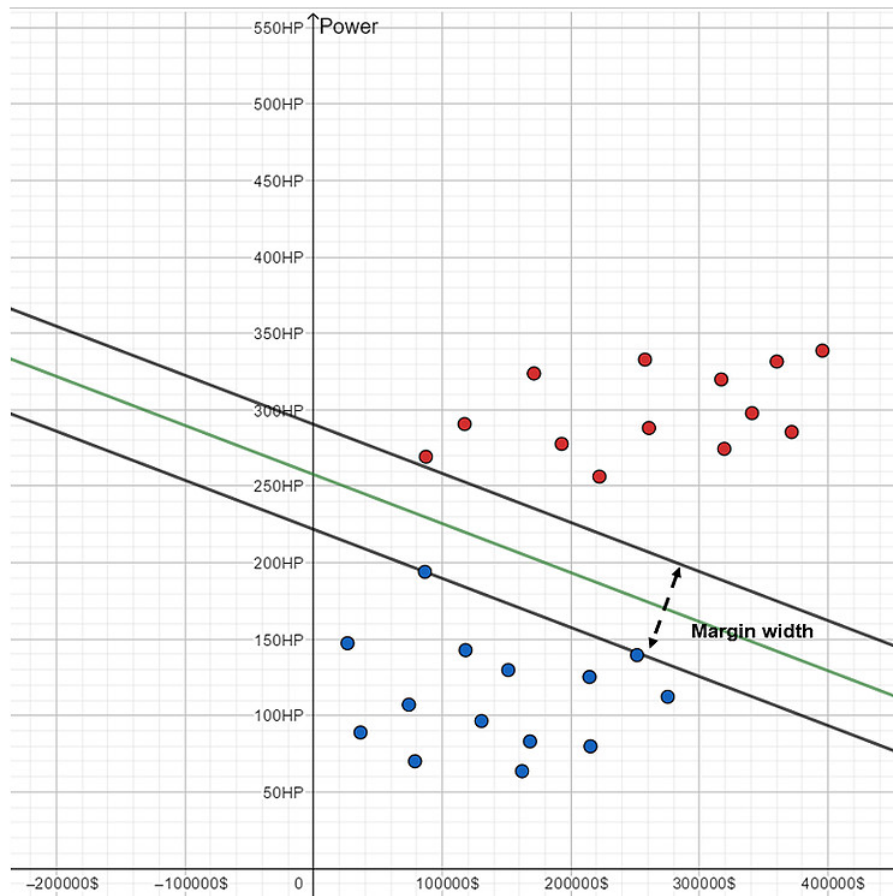
4.2 MÁQUINA DE VETORES DE SUPORTE - SVM

Algoritmos de máquina de vetores de suporte, ou *Support Vector Machine* são classificadores supervisionados, capazes de identificar classes analisando características extraídas de imagens ou dados. As SVM reconhecem características específicas para cada classe, dessa forma encontrando a diferenciação entre os objetos analisados, e traça uma linha entre os objetos das duas classes, separando-os. São chamadas vetores de suporte pois existem infinitas retas possíveis entre objetos da classe A e os objetos da B, e o algoritmo SVM cria os vetores que ajudam a encontrar aquele vetor que mais se distancia das duas classes analisadas, assim diminuindo a chance de analisar uma amostra erroneamente.

Um exemplo de classificador SVM é mostrado na figura 4.4 simboliza o vetor definido para classificar tipos de carro entre esportivo e não esportivo, o algoritmo utiliza de características potência e preço para identificar a qual classe cada amostra analisada pertence. Os vetores de suporte estão identificados, assim como a distância entre eles e o vetor principal.

O código que utiliza a estrutura inicial para treinar e a estrutura final para classificar as amostras a partir do resultado obtido pelo algoritmo treinado, é mostrado abaixo, os códigos são diferentes, pois as funções se diferem para algoritmos SVM que classificam em duas classes - funções *svmtrain* para treinamento e *svmclassify* para classificação - e SVM em múltiplas classes - funções *fitcecoc* para treinamento e

FIGURA 4.4 – Algoritmo de Vetores de Suporte



Fonte: Machine Unlearned

predict para classificação:

```
svmStruct = svmtrain(hog_image,groupTraining,'options',options);

for k = 1 : 2*(TestSamples-2)
    person(k,:) = svmclassify(svmStruct,Xnew(k,:), 'ShowPlot',true);
end

acceptanceRate = sum(strcmp(groupTest,person)) / (2*(TestSamples-2))
```

(4.1)

```
svmStruct = fitcecoc(hog_image,groupTraining);

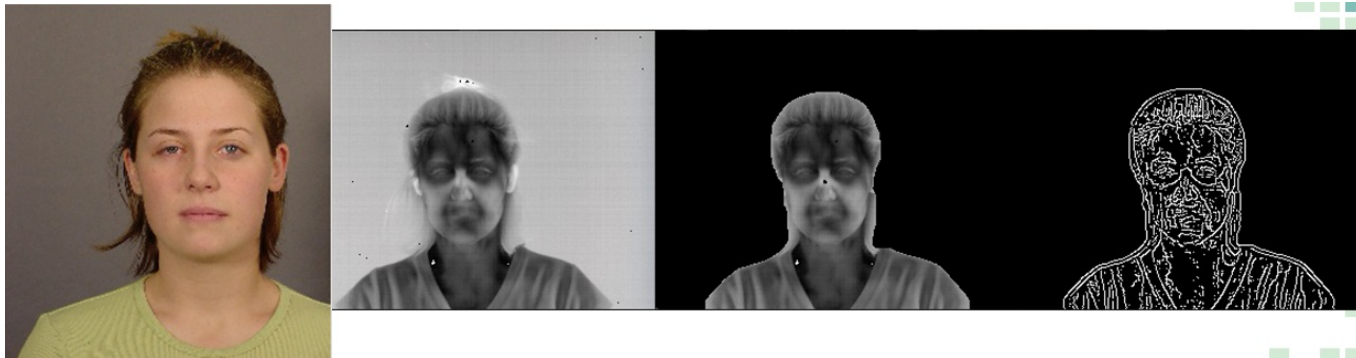
for k = 1 : 3*TestSamples
    person(k,:) = predict(svmStruct,Xnew(k,:));
end
```

(4.2)

5 RESULTADOS E DISCUSSÕES

Analisando o desenvolvimento do presente trabalho, é possível dividi-lo em um detector de veias, desenvolvido em Python e um classificador, desenvolvido e simulado em MATLAB. O método utilizado foi o Canny 50,50. A resposta apresentada nas figuras 5.1 e 5.2 ainda continha diversos erros, contendo dados excedentes aos necessários e impossibilitando o trabalho futuro de classificação. Retrabalhando o código de detecção, a fim de mapear corretamente as principais veias da face, e eliminar os ruídos, o resultado final do extrator é mostrado nas figuras 5.3 e 5.4.

FIGURA 5.1 – Imagem Binária Inconclusiva



Fonte: Autora

FIGURA 5.2 – Imagem Binária Inconclusiva



Fonte: Autora

Foi necessário extrair características das imagens utilizadas para treinamento e também das utilizadas no teste para validação do classificador. As porções de código

FIGURA 5.3 – Detecção de Veias - Homem A



Fonte: Autora

FIGURA 5.4 – Detecção de Veias - Homem B



Fonte: Autora

abaixo mostram como é feita a extração de características e a criação da estrutura de classes analisada em MATLAB.

```

for k = 1 : samples

    imgA = imread(strcat(PATH,'HomemA_ ',num2str(k),').tiff');
    grayImgA = 255 * uint8(imgA);
    imgA_RGB = cat(3, grayImgA, grayImgA, grayImgA);

    imgB = imread(strcat(PATH,'HomemB_ ',num2str(k),').tiff');
    grayImgB = 255* uint8(imgB);
    imgB_RGB = cat(3, grayImgB, grayImgB, grayImgB);

    imgC = imread(strcat(PATH,'MulherA_ ',num2str(k),').tiff');
    grayImgC = 255* uint8(imgB);
    imgC_RGB = cat(3, grayImgC, grayImgC, grayImgC);

    hog_image(k,:) = extractHOGFeatures((imgA_RGB),'CellSize',[4 4]);
    hog_image(k+samples,:) = extractHOGFeatures((imgB_RGB),'CellSize',[4 4]);
    hog_image(k+(2*samples),:) = extractHOGFeatures((imgC_RGB),'CellSize',[4 4]);

    groupTraining(k,1) = 'HomemA';
    groupTraining(samples + k,1) = 'HomemB';
    groupTraining(2*samples + k,1) = 'MulherA';

end

```

(5.1)

```

for k = 1 : samples

    imgA = imread(strcat(PATH, 'HomemA_ ', num2str(k), '.tiff'));
    grayImgA = 255 * uint8(imgA);
    imgA_RGB = cat(3, grayImgA, grayImgA, grayImgA);

    imgB = imread(strcat(PATH, 'HomemB_ ', num2str(k), '.tiff'));
    grayImgB = 255 * uint8(imgB);
    imgB_RGB = cat(3, grayImgB, grayImgB, grayImgB);

    imgC = imread(strcat(PATH, 'MulherA_ ', num2str(k), '.tiff'));
    grayImgC = 255 * uint8(imgB);
    imgC_RGB = cat(3, grayImgC, grayImgC, grayImgC);

    hog_image(k,:) = extractHOGFeatures((imgA_RGB), 'CellSize', [4 4]);
    hog_image(k+samples,:) = extractHOGFeatures((imgB_RGB), 'CellSize', [4 4]);
    hog_image(k+(2*samples),:) = extractHOGFeatures((imgC_RGB), 'CellSize', [4 4]);

    groupTraining(k,1) = 'HomemA';
    groupTraining(samples + k,1) = 'HomemB';
    groupTraining(2*samples + k,1) = 'MulherA';
end

```

(5.2)

O treinamento do algoritmo utilizou 4 amostras diferentes por indivíduo, e o teste de classificação foi feito utilizando outras duas amostras desses indivíduos, aleatoriamente importadas pelo MATLAB. O resultado do treinamento teve uma taxa de 100% de acerto na classificação de dois indivíduos, e 61,67% de acerto para a classificação de três indivíduos distintos. Um resultado bastante satisfatório considerando que os dados de entrada para o algoritmo SVM foram as imagens processadas durante o desenvolvimento deste projeto, e que o classificador conseguiu convoluir de forma a classificar indivíduos diferentes.

6 CONCLUSÃO

As principais dificuldades de visão computacional encontradas ao iniciar um desenvolvimento semelhante à capacidade de visão dos seres humanos são aquisição e armazenamento de uma base de dados bastante qualificada, e o desenvolvimento de algoritmos que contenham técnicas de adaptação às condições muito variadas de ambiente, clima e iluminação. Também o procedimento de reconhecimento exige um processamento pesado e de altíssima velocidade para resultados satisfatórios. Para o desenvolvimento desse projeto, o banco de imagens térmicas utilizado foi concedido pela Universidade de Notre Dame através de um *Release Agreement* para seu uso.

Após a preparação do sistema para trabalhar com imagens digitais, isto é instalação de compiladores e bibliotecas específicas que foram utilizadas ao longo desse trabalho, o desenvolvimento de um algoritmo de reconhecimento de face em Python se iniciou. Técnicas específicas de processamento digital de imagens foram utilizadas, buscando a obtenção da região de interesse em cada uma das imagens do banco. Essa região de interesse é a área da face, e com a imagem segmentada nessa região é possível iniciar o trabalho de extração das características, o mapeamento das veias faciais.

Os objetivos ao desenvolver esse projeto foram atingidos, como a extração de veias dos indivíduos a partir das imagens térmicas, considerando a importância da obtenção de informações corretas, atentando para as principais artérias e veias faciais, deixando de lado pequenas ramificações que podem ser encontradas como falsos positivos, para que fosse possível a classificação.

Também foi desenvolvido no trabalho futuro um estudo aprofundado e compreensão de técnicas de *machine learning* a fim de comparar e aplicar a melhor técnica para o projeto em questão, de reconhecimento de indivíduos, o método utilizado foi o SVM e o resultado foi satisfatório. A classificação dos dados foi realizado de maneira inteligente podendo segregar imagens do banco por indivíduo, e por fim o reconhecimento dos indivíduos com uma taxa de erro aceitável.

Com isso, o trabalho proporcionou aprendizado para a aluna em diversas áreas da tecnologia, de grande importância para o mercado de trabalho, tendo em vista a necessidade atual do reconhecimento preciso de identidade.

REFERÊNCIAS

- ABRÃO, R. A. **Bradesco lança caixa eletrônico que lê veias da mão**. 2008. <<https://oglobo.globo.com/economia/bradesco-lanca-caixa-eletronico-que-le-veias-da-mao-4552602>>. Acesso em: jun. 2018. Citado na página 15.
- BAKKER, G. **Color Coding and the RGB Color Space**. 2004. <<http://geraldbakker.nl/psnumbers/rgb-explained.html>>. Acesso em: mar. 2018. Citado na página 11.
- BARROW, J. T. H. **Computational Vision**. *Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics*, Vol. 14, No. 1, IEEE, 1981. Citado na página 11.
- BIOINFO. **Espectro Eletromagnético**. 2012. <<https://bioinformaticacc.wordpress.com/2012/05/31/espectro-eletromagnetico/>>. Acesso em: mar. 2018. Citado 2 vezes nas páginas 7 e 10.
- BOURLAI ARUN ROSS, C. C. L. H. T. **A Study on Using Mid-Wave Infrared Images for Face Recognition**. *West Virginia University, Proceedings of SPIE*, 2017. Citado 2 vezes nas páginas 7 e 15.
- CAMPOS SERGIO BAMPI, A. S. T. Sistema de identificação de placas por processamento automático de imagens. <http://www.iberchip.net/VII/cdnav/pdf/49.pdf>, 2010. Citado na página 10.
- FILHO O. M.; NETO, H. V. **Processamento digital de imagens**. [S.l.]: Brasport, 1999. Citado 4 vezes nas páginas 6, 9, 12 e 23.
- GONZALEZ, R. E. W. R. C. **Processamento de Imagens Digitais**. [S.l.]: Edgard Blücher LTDA, 2000. Citado 3 vezes nas páginas 6, 20 e 26.
- GURU, M. L. **Image Filtering**. 2017. <http://machinelearningguru.com/computer_vision/basics/convolution/image_convolution_1.html> .Acesso em : abr.2018.Citadonapágina21.
- HITACHI. **Finger Vein Authentication Technology**. 2012. <<http://www.hitachi.co.jp/products/it/veinid/global/introduction/fingevein.html>>. Acesso em: mar. 2018. Citado na página 14.
- JAIN ARUN ROSS, S. P. T. K. **An Introduction to Biometric Recognition**. *Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics*, Vol. 14, No. 1, IEEE, 2004. Citado 3 vezes nas páginas 11, 12 e 13.
- LOPES, E. C. **Detecção de Faces e Características Faciais**. *Pontifícia Universidade Católica do Rio Grande do Sul*, PUCRS, 2010. Citado na página 14.
- MISHRA KANDERP NARAYAN MISHRA, A. A. K. N. Veins based personal identification systems: A review. *International Journal of Intelligent Systems Technologies and Applications*, 2016. Citado na página 6.

OLIVEIRA, N. J. L. Marcelo de A. **Reconnection of Fingerprint Ridges Based on Morphological Operators and Multiscale Directional Information**. *UNICAMP - State University of Campinas*, IEEE, 2004. Citado na página 12.

PEDRINI, W. R. S. H. **Análise de Imagens Digitais**. [S.l.]: Thomson Learning, 2008. Citado na página 10.

SAVVIDES, D. M. **Introduction to Biometric Recognition Technologies and Applications**. 2010. <<https://users.ece.cmu.edu/>>. Acesso em: mar. 2018. Citado na página 11.

WHAT-WHEN-HOW. **Face Alignment Models (Face Image Modeling and Representation) (Face Recognition)**. 2016. <<http://what-when-how.com/face-recognition/>>. Acesso em: mai. 2018. Citado 2 vezes nas páginas 6 e 14.