

SONIA ISOLDI MARTY GAMA MÜLLER

**COMPARAÇÃO ENTRE OS MÉTODOS DE MÁXIMA
VEROSSIMILHANÇA, DISTÂNCIA MÍNIMA E O MÉTODO
DE FISHER PARA RECONHECIMENTO DE PADRÕES
EM IMAGENS COLORIDAS.**

Dissertação apresentada ao Curso de Pós-Graduação em Ciências Geodésicas do Setor de Tecnologia da Universidade Federal do Paraná, como requisito parcial à obtenção do Título de Mestre em Ciências.

Orientador: Prof. Dr. Quintino Dalmolin.

Co-orientador: Prof. Hideo Araki MSc.

UNIVERSIDADE FEDERAL DO PARANÁ

1997

**“COMPARAÇÃO ENTRE O MÉTODO DE MÁXIMA
VEROSSIMILHANÇA, DISTÂNCIA MÍNIMA E O
MÉTODO DE FISHER PARA RECONHECIMENTO
DE PADRÕES EM IMAGENS COLORIDAS”.**

POR

SONIA ISOLDI MARTY GAMA MÜLLER

Dissertação aprovada como requisito parcial do grau de Mestre no Curso de Pós-Graduação em Ciências Geodésicas da Universidade Federal do Paraná, pela Comissão formada pelos professores:



Prof. Dr. QUINTINO DALMOLIN - Orientador e Presidente



Prof. HIDEO ARAKI MSc - Co-Orientador



Prof. Dr. JAIR MENDES MARQUES- Membro



Prof. Dr. OLGA REGINA PEREIRA BELLON- Membro

“Assim como luz e trevas são as duas forças infinitas que devoram tudo que se manifesta, todas as coisas vivem, nestas duas, em eterna criação e dissolução. Assim se inflama, na profundidade incolor e transparente, a cor; na medida em que a cor se move mais intensamente dentro dela mesma, ela se incendeia, cada vez mais grandiosa e luminosamente. No momento em que a cor submerge no profundo espaço incolor, nessa profundidade viva, a mais delicada iluminação novamente inflama as cores, de modo que ressurgem como um acorde há muito conhecido. Assim a cor aparece igual ao fulgor das estrelas fixas, partindo, com que em ondas majestosas, ao júbilo fulgurante da suma claridade, rumo ao abismo sonoro da treva infindável, no ritmo de sua respiração grandiosa, em sono cada vez mais profundo, até que, em tempos vindouros a luz nele incendeie o ardor da manhã com infinito esplendor.”

Philipp Otto Runge

À minha Família

AGRADECIMENTOS

Dedico este trabalho a todos aqueles que direta ou indiretamente contribuíram para o seu desenvolvimento e elaboração.

Primeiramente, agradeço ao Prof. Dr. Antonio Maria Garcia Tommaselli, pela sua contribuição no tema desta dissertação.

Agradeço em especial aos meus orientadores Prof. Quintino e Prof. Hideo, pela compreensão, estímulo e valiosa contribuição didático profissional, que de forma decisiva possibilitaram a elaboração, análise e formulação desta dissertação de mestrado.

Agradeço em especial também a aquelas colegas que de uma forma pessoal participaram em momentos decisivos deste trabalho, aos quais não poderia furtar-me de mencionar nominalmente:

- Anselmo ao incentivo para assumir este desafio profissional e ao valioso apoio de fundamentação didática;
- Carmen ao apoio espiritual e de estímulo;
- Jair à colaboração para o correto direcionamento da pesquisa e suas valiosas informações;
- Lígia aos conselhos, sugestões e apoio decisivo para a elaboração desta dissertação;
- Milioni ao encabeçar as atividades didáticas, permitindo-me maior envolvimento nos trabalhos desta dissertação;
- Nelcimar pelo apoio e coleguismo durante a fase teórica do mestrado;
- Wilson às valiosas contribuições na formulação deste trabalho.

Aos colegas Professores do Departamento de Estatística da Universidade Federal do Paraná, que ao assumir os trabalhos didáticos que

realizava, permitiram a minha dedicação exclusiva aos trabalhos de elaboração desta dissertação de mestrado.

Agradeço também aos colegas Professores do ITA que diariamente conviveram os últimos meses deste meu trabalho e transmitiram positivismo e motivação para sua conclusão.

Agradeço de forma carinhosa aos meus filhos, que souberam em muitos momentos, na inocência de sua solidária abstinência ao carinho de uma mãe, contribuir para que pudesse desenvolver e realizar este trabalho.

Ao meu marido, companheiro decisivo pelo apoio e motivação despertados nos momentos de incerteza, bem como pela compreensão no compartilhamento de meus objetivos profissionais, procurando ser solidário e dando total apoio à realização de minhas aspirações profissionais e pessoais.

À minha mãe, sempre presente em todos os momentos de minha vida, pelo seu carinho e paciência.

Agradeço por último de forma não menos importante, aos meus sogros pelo apoio familiar, bem como pelo voto de confiança que depositaram neste trabalho e em minha capacidade profissional.

SUMÁRIO

1. INTRODUÇÃO.....	01
2. CONCEITOS BÁSICOS.....	05
3. MÉTODOS E TESTES ESTATÍSTICOS.....	14
3.1. MÉTODOS ANALISADOS.....	14
3.1.1. MÉTODO DE MÁXIMA VEROSSIMILHANÇA.....	14
3.1.2. MÉTODO DE DISTÂNCIA MÍNIMA.....	18
3.1.3. MÉTODO DE FISHER.....	20
3.2. TESTES ESTATÍSTICOS.....	29
3.2.1. AVALIAÇÃO DO DESEMPENHO DE CLASSIFICAÇÃO.	30
3.2.2. TESTE PARA DUAS PROPORÇÕES.....	35
4. METODOLOGIA.....	37
4.1. CONSIDERAÇÕES GERAIS.....	39
4.2. ALGORITMO.....	45
5. ANÁLISE DOS RESULTADOS.....	50
5.1. 1o. CASO.....	50
5.2. 2o. CASO.....	63
6. CONCLUSÕES E RECOMENDAÇÕES.....	70
7. REFERÊNCIAS BIBLIOGRÁFICAS.....	72
ANEXO I.....	75

ÍNDICE DE FIGURAS

Figura 2.1 - Espectro Eletromagnético.....	05
Figura 2.2 - Concepção de Imagem Digital Colorida.....	07
Figura 2.3 - Cubo do RGB.....	08
Figura 2.4 - Processo Aditivo de Composição de Cores.....	09
Figura 2.5 - Funções Densidade de Probabilidade Definida por Máxima Verossimilhança.....	11
Figura 2.6 - Representação do Método de Distância Mínima.....	11
Figura 2.7.- Representação do Procedimento de Fisher para Duas Populações.....	12
Figura 4.1 - Etapas do Processo de Comparação dos Métodos.....	39
Figura 4.2 - Imagem Gerada para Identificação de Letras Verdes.....	43
Figura 4.3 - Foto do Mandril com parte Seleccionada para Classificação.....	44
Figura 4.4 - Opções do Algoritmo de Controle.....	45
Figura 4.5 - Opções da Entrada de Cores.....	46
Figura 4.6 - Opções do Tipo de Amostra.....	46
Figura 4.7 - Opções do No. de Cores.....	46
Figura 4.8 - Opções da Análise Discriminante.....	47
Figura 4.9 - Análise Visual do Desempenho dos Métodos.....	47
Figura 4.10 - Opções da Saída do Programa.....	48

Figura 4.11 - Esquema do Programa.....	49
Figura 5.1 -Análise Visual da Comparação dos Métodos da Imagem de Letras Considerando Componentes Dependentes.....	61
Figura 5.2 - Análise Visual da Comparação dos Métodos de Parte da Foto Digitalizada Considerando Componentes Dependentes.....	61
Figura 5.3 -Análise Visual da Comparação dos Métodos da Imagem de Letras Considerando Componentes Independentes.....	63
Figura 5.4 - Análise Visual da Comparação dos Métodos de Parte da Foto Digitalizada Considerando Componentes Independentes.....	69

ÍNDICE DE TABELAS.

Tabela 3.1 - Tabela de Contingência do Método de Máxima Verossimilhança com Amostras de Médias Próximas.....	31
Tabela 4.1 - Cores Definidas para Classificação.....	39
Tabela 5.1 - Percentual de Erros de Classificação por Tipo de Média Amostral Considerando Dependência entre as Componentes.....	52
Tabela 5.2 - Percentual de Erros de Classificação por Tipo de Variância Amostral Considerando Dependência entre as Componentes.....	54
Tabela 5.3 - Total e Percentual de Erros de Classificação por No. de Cores Discriminadas Simultaneamente Considerando Dependência entre as Componentes	57
Tabela 5.4 - Tempo Médio de Processamento por Método e No. de Cores Simultaneamente Discriminadas Considerando Dependência entre as Componentes.....	58
Tabela 5.5 - Tabela de Contingência - Método de Máxima Verossimilhança Considerando Dependência entre as Componentes da Imagem de Letras.....	59
Tabela 5.6 - Tabela de Contingência-Método de Distância Mínima Considerando Dependência entre as Componentes da Imagem de Letras..	60
Tabela 5.7 - Tabela de Contingência-Método de Fisher Considerando Dependência entre as Componentes da Imagem de Letras.....	60
Tabela 5.8 - Tabela de Contingência - Método de Máxima Verossimilhança Considerando Dependência entre as Componentes para Imagem Digitalizada.....	62
Tabela 5.9 - Tabela de Contingência- Método de Distância Mínima Considerando Dependência entre as Componentes para Imagem Digitalizada.....	62

- Tabela 5.10 - Tabela de Contingência - Método de Fisher Considerando Dependência entre as Componentes para Imagem Digitalizada...62
- Tabela 5.11 - Percentual de Erros de Classificação por Tipo de Média Amostral Considerando Independência entre as Componentes.....64
- Tabela 5.12 - Percentual de Erros de Classificação por Tipo de Variância Amostral Considerando Independência entre as Componentes. 65
- Tabela 5.13 - Total e Percentual de Erros de Classificação por No. de Cores Discriminadas Simultaneamente Considerando Independência entre as Componentes.....66
- Tabela 5.14 - Tempo Médio de Processamento por Método e No. de Cores Simultaneamente Discriminadas Considerando Independência entre as Componentes.....67
- Tabela 5.15 - Tabela de Contingência - Método de Máxima Verossimilhança Considerando Independência entre as Componentes da Imagem Digitalizada.....68
- Tabela 5.16 - Tabela de Contingência-Método de Distância Mínima Considerando Independência entre as Componentes da Imagem Digitalizada.....68
- Tabela 5.17 - Tabela de Contingência - Método de Fisher Considerando Independência entre as Componentes da Imagem Digitalizada.....68

RELAÇÃO DOS PRINCIPAIS SÍMBOLOS E ABREVIATURAS. LISTA DE SÍMBOLOS

π_i - classe (população) a ser separada pelo método (no caso refere-se as cores a serem discriminadas).

$p(x)$ - probabilidade de x ocorrer.

$p(\pi_i)$ - probabilidade da classe i ocorrer.

$p(\pi_k/x)$ - probabilidade de ocorrer a classe k dado que x já tenha ocorrido.

$p(x/\pi_k)$ - probabilidade de ocorrer x dado que a classe k já tenha ocorrido.

$g_i(x)$ - função discriminante de x da classe i .

μ_i - vetor médio populacional da classe i .

Σ_i - matriz de covariância populacional da classe i .

\bar{x}_i - vetor médio amostral da classe i .

S_i - Matriz de covariância amostral da classe i .

l' - vetor de coeficientes da equação linear de Fisher.

$E(Y/\pi_i)$ - esperança de Y dado que ocorreu a classe i .

$Var(X)$ - variância de X .

$Cov(X)$ - covariância de X .

S_c - matriz de covariância conjunta.

$\bar{\mu}$ - vetor médio combinado das classes.

B_0 - Soma dos produtos cruzados entre as classes .

$\bar{\mu}_Y$ - média global para todas as classes.

$\bar{\sigma}_Y$ - variância de Y.

Σ - matriz de covariância conjunta.

$\Sigma^{1/2}$ - matriz raiz quadrada simétrica.

λ - autovalores.

e - autovetores.

$\hat{\beta}_0$ - estimativa da matriz soma dos produtos cruzados entre os grupos.

W - matriz soma de produtos cruzados dentro dos grupos amostrais.

$\hat{\Sigma}$ - estimativa da matriz de covariância conjunta.

$\hat{\lambda}_i$ - autovalores estimados.

\hat{e}_i - autovetores estimados.

σ^2 - variância populacional.

s^2 - variância amostral.

I - matriz identidade.

RESUMO

Uma das grandes preocupações no processo de automação em Fotogrametria está relacionado com os algoritmos de reconhecimento de padrões. Apesar da metodologia ter avançado significativamente nos últimos anos, principalmente no que se refere ao tom de cinza, o reconhecimento de padrões de cores reais em imagens coloridas ainda se encontra em fase de desenvolvimento. Neste sentido objetiva-se realizar uma série de ensaios com dados (cores) controlados e também com dados reais para comparar a performance dos métodos estatísticos de Máxima Verossimilhança, Distância Mínima e de Fisher para reconhecimento das cores reais. A comparação entre os métodos estatísticos foi feita controlando o processo de classificação, ou seja, utilizando amostras determinísticas para treinamento das funções discriminantes e criando imagens para classificação no *software* (MATLAB). Com o intuito de que a comparação pudesse ser feita sem controle da imagem a ser classificada e das amostras, utilizou-se uma imagem retirada de uma foto digitalizada, para verificar a performance dos métodos frente a um caso real. A análise de desempenho dos métodos foi observada pela estatística *kappa* e a comparação pelo teste Z apropriado. Os resultados indicam que o Método de Distância Mínima foi o melhor para classificação de cores reais sob todos os aspectos estudados.

ABSTRACT

One of the main issues in the photogrammetry automation process is related to pattern recognition algorithms. Although there were significant advances in the last years, mainly in regard to the gray tones, color pattern recognition in colored images is still in a developmental stage. In this sense, a series of tests with controlled data (colors) and also with real data is made to compare the performance of the statistical methods: Maximum Likelihood, Minimum Distance, and Fisher's method in their ability to recognize real colors. The comparison among the statistical methods was performed by controlling the classification process, that is, using deterministic samples for the training of discriminant functions and test images created by a software (MATLAB). For the case of real images, the statistical methods were compared without controlling both the sample and the image by the utilization of an image extracted from a digitized picture. The methods performance was analyzed using the kappa statistic and the Z test. The results indicated that the Minimum Distance Method was the best method to classify real colors under all assumptions.

1. INTRODUÇÃO.

Durante muitos anos, o meio mais eficaz de identificar objetos em imagens coloridas tem sido o visual. Porém, o olho humano pode identificar apenas duas mil combinações diferentes de cores, enquanto que o computador pode representar mais de dezesseis milhões destas combinações (GARCIA 1982).

Na atual visão computacional, a capacidade de reconhecimento de um objeto em uma imagem é bastante significativa para a obtenção de informações, visto que algumas propriedades como a geometria, a textura, a sombra, a posição do objeto e principalmente a cor, são essenciais para o reconhecimento e podem ser descritos em meio digital.

O ser humano tem uma capacidade fantástica de trabalhar com estas várias propriedades ao mesmo tempo, porém a geometria do objeto e a cor são as informações que possuem maior influência sobre o cérebro, ao se tentar detectar um objeto numa cena. Com a finalidade de ilustrar, pode-se imaginar uma cena onde existe um cesta de frutas que contém pêras, uvas, laranjas e maçãs. Ao se tentar identificar as laranjas contidas nesta cesta, pensa-se em sua forma ovalada e sua cor alaranjada. As propriedades, mencionadas acima, são importantes para o reconhecimento, mas a cor supera em alguns casos as outras, pois mesmo a uma distância relativamente grande torna-se possível identificar as laranjas, procurando-se a cor alaranjada. Assim, ao se extrair uma cor de uma imagem, e se esta for única para determinado objeto, pode-se obter automaticamente a sua geometria.

Para o reconhecimento de padrões em imagens coloridas, é de vital importância a extração de suas cores. Não se descarta, no entanto, a importância da geometria e das outras propriedades dos objetos em uma imagem, que se trabalhadas em conjunto trazem um melhor efeito no reconhecimento. A opção de trabalhar somente a propriedade cor, pode levar a racionalização dos custos decorrentes do tempo de processamento e dos custos de se obter mais informações do objeto a ser reconhecido, pois existe uma crescente necessidade, nos últimos tempos, de reconhecimento de padrões em tempo real (*on-line*).

Técnicas estatísticas de reconhecimento de padrões tem sido exploradas para classificação de imagens em sensoriamento remoto, com aplicações na exploração mineral, agricultura, reflorestamento, astronomia, e também na medicina, arquitetura, robótica (visão de máquina), etc. (ARGIALAS & HARLOW, GREEN; SONKA, HLAVAC & BOYLE; GROOVER, WEISS, NAGEL & ODREY).

Dos métodos estatísticos utilizados, observa-se que o de Máxima Verossimilhança é o mais popular devido principalmente a sua performance, sendo, por esta razão, que muitas das novas técnicas apresentadas são comparadas com este método (BOLSTAD & LILLESAND, 1990).

Encontrou-se, no entanto, uma preocupação em aprimorar o Método de Máxima Verossimilhança como nos artigos de BOLSTAD & LILLESAND (1990) e DEUSEN (1995), onde o primeiro apresenta uma modificação do algoritmo para melhorar a sua velocidade de processamento e o segundo melhorar a performance.

Pode-se citar outros métodos para reconhecimento digital de um objeto, como no caso de classificação supervisionada (classes conhecidas) os métodos de Distância Mínima, do Paralelepípedo e de Fisher e para o caso não-supervisionado (classes desconhecidas) os métodos de agrupamento (clustering) que podem ser feitos com base na distância ou na similaridade (RICHARDS; LILLESAND & KIEFER; SONKA, HLAVAC & BOYLE; FOODY, McCULLOCH & YATES; HUANG; MAUSAL; PEDDLE e JOHNSON & WICHERN).

Recentemente a atenção focaliza-se na utilização de redes neurais artificiais para classificação de imagens e que segundo FOODY, McCULLOCH & YATES (1995) tem apresentado alguma melhoria na classificação e diferenças pouco significativas em tempo de processamento.

Com a extensa aplicação do reconhecimento de imagens coloridas, principalmente no campo da robótica, onde os cientistas vem proporcionando avanços substanciais a cada dia e com a tecnologia atual de câmeras digitais coloridas e o amplo uso das câmeras de vídeo, faz-se necessário o desenvolvimento de estudos no que diz respeito a cores reais.

A Sociedade Internacional de Fotogrametria e Sensoriamento Remoto recomenda estudos de métodos e técnicas para o reconhecimento de padrões, devido ao aumento na demanda de aplicações na automação, principalmente em processos de inspeção de qualidade onde devido a rapidez dos processadores de dados atuais está sendo possível que câmeras substituam a visão humana.

As aplicações na robótica (visão de máquina) subdividem-se em três grandes categorias: inspeção, identificação e servomovimentação visual. Na primeira, os objetos, por meio da visão de máquina, são verificados quanto as suas falhas; a segunda categoria está envolvida com aplicações nas quais a finalidade do sistema de visão é reconhecer e classificar um objeto. Na última, servomovimentação, tem por finalidade dirigir as ações do robô baseado em informações visuais recebidas. Pensa-se que aplicações de

reconhecimento de cores para as três categorias poderiam servir nas seguintes situações: inspecionar cores em objetos ou uniformidade das mesmas; selecionar e classificar objetos de acordo com sua cor (para ilustrar, pode-se pensar em separar laranjas maduras (alaranjadas) de laranjas não-maduras (verdes) e colocá-las em caixas distintas); e ainda, poder-se-ia direcionar um robô através de uma faixa vermelha para determinado serviço e outra amarela para uma segunda atividade e assim por diante.

O aprofundamento destes estudos implica na necessidade de se avaliar se as técnicas estatísticas usualmente utilizadas para reconhecimento de imagens em Sensoriamento Remoto poderiam ser transportadas diretamente para o estudo de cores reais. Mais especificamente, o objetivo deste trabalho foi comparar os métodos estatísticos de Máxima Verossimilhança, Distância Mínima e de Fisher com a finalidade de verificar qual desempenha melhor a função de identificar objetos em imagens digitais coloridas.

Neste trabalho, procurou-se abordar no capítulo 2 alguns conceitos básicos que fornecem subsídios a um melhor entendimento da aplicação dos métodos, sendo alguns de muita importância neste trabalho, como por exemplo: pixel, imagem digital colorida, sistema RGB, processo aditivo de composição de cores, reconhecimento de padrão, análise discriminante, classificação, etc.; bem como apresentou-se também uma descrição sucinta dos métodos utilizados e das estatísticas que possibilitaram uma comparação entre eles.

Os métodos selecionados e os testes estatísticos utilizados apresentam-se descritos com detalhes no capítulo 3. O método de Máxima Verossimilhança, descrito no item 3.1.1, é concebido a partir de uma função de penalidade, onde se introduz uma condição de normalidade chegando-se assim à regra de decisão do método. Partindo-se de uma restrição referente à variância dos dados analisados no método de Máxima Verossimilhança chega-se à função discriminante do Método de Distância Mínima, apresentada no item 3.1.2. O método de Fisher é deduzido a partir da linearização das observações, escolhendo-se a melhor combinação linear que represente os dados e utilizando a distância quadrática chega-se às regras de alocação estabelecidas por ele, mostrada no item 3.1.3. Os testes estatísticos utilizados no trabalho são apresentadas no item 3.2, onde em sua introdução descreve-se sucintamente a concepção dos testes e com mais detalhes o teste de avaliação de desempenho de classificação (item 3.2.1) que determina através de uma tabela chamada de tabela de contingência, uma estatística *kappa*, que nada mais é que um índice de bom desempenho de determinado método, podendo ser comparado a um *kappa* de outro método através de um

teste Z. Outro teste, descrito no item 3.2.2, é o teste para duas proporções, utilizado nas tabelas de percentuais de erros de classificação para os vários experimentos analisados neste trabalho.

O processo de comparação dos métodos pode ser visto detalhadamente no capítulo 4, onde são relatados todos aspectos analisados na comparação dos métodos, e é descrito o algoritmo de comparação composto de uma rotina de controle que comanda várias sub-rotinas. Incluiu-se também a apresentação das imagens de saída em tela das rotinas e sub-rotinas de todo o processo de análise.

Os resultados apresentados no capítulo 5 aparecem divididos em dois casos propostos no capítulo 4, onde no primeiro se analisou os dados da forma como os métodos foram concebidos, isto é, sem sofrer nenhuma modificação; no segundo caso, se modificou os métodos considerando-se que as componentes apresentassem independência entre si. Para cada caso são mostrados os resultados obtidos frente a alguns aspectos diferentes que serão chamados de experimentos, sendo que os experimentos 1 e 2 referem-se as características das amostras utilizadas no treinamento do discriminante, que são média e variância respectivamente; o experimento 3 verifica o comportamento dos métodos considerando o número de cores simultaneamente classificadas; o experimento 4 mede o tempo de processamento de cada método identificando o mais rápido; o experimento 5 faz uma análise de uma imagem que contém as 15 cores propostas no trabalho donde se deseja classificar apenas os tons de verde e o experimento 6 analisa uma imagem captada de parte de uma foto digitalizada para que se possa verificar o método de melhor performance em uma condição prática.

Finalmente, no capítulo 6, apresenta-se o método que obteve melhor resultado para o reconhecimento específico de cores em uma imagem, além de outras conclusões e recomendações decorrentes dos resultados obtidos.

2. CONCEITOS BÁSICOS.

Neste capítulo, apresenta-se os conceitos necessários ao entendimento de todo o processo de comparação dos métodos, incluindo desde a definição de imagem, até uma descrição sucinta dos métodos de Máxima Verossimilhança, Distância Mínima e de Fisher para análise de reconhecimento de cores.

Segundo GARCIA (1982), uma **imagem de intensidade luminosa** é o resultado do registro da energia refletida, emitida e/ou transmitida das diferentes partes do espectro eletromagnético, assim cada objeto reflete, absorve ou irradia seletivamente a energia segundo sua própria estrutura e composição.

O registro de uma imagem é feita através de sensores que captam a energia eletromagnética. O olho humano é considerado um sensor apto a detectar a radiação refletida de uma pequena parte do espectro eletromagnético compreendida aproximadamente entre 0,4 a 0,7 μm , chamada de **faixa do visível** (Figura 2.1).

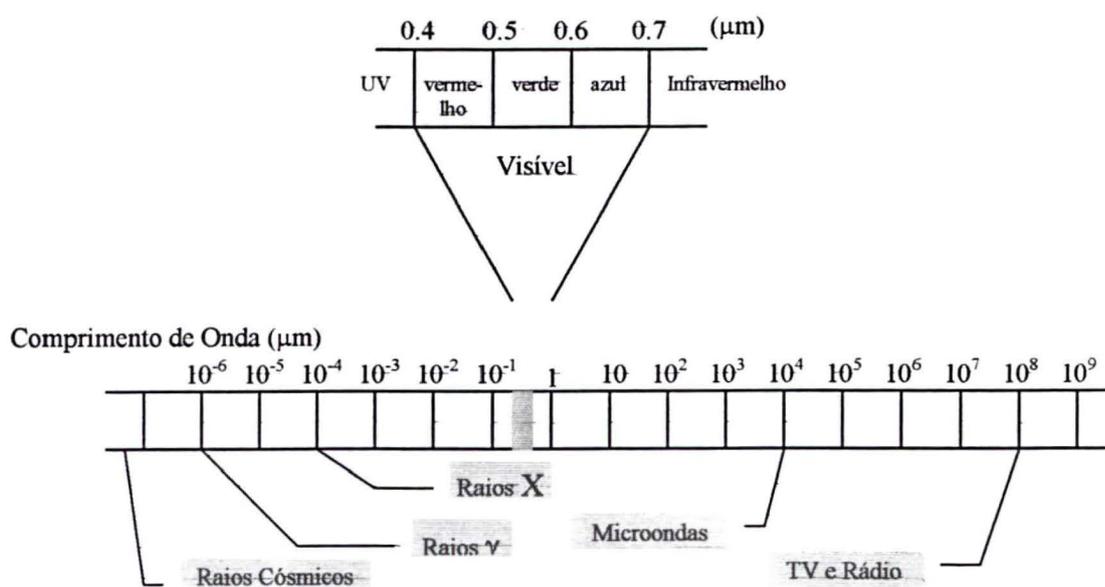


Figura 2.1. Espectro Eletromagnético

Fonte: reproduzida de LILLESAND & KIEFER (1994)

Nos sistemas de imageamento eletro-óticos, segundo NOVO (1992), os dados são registrados em forma de sinais elétricos e compostos por dois componentes básicos: o sistema ótico e o detetor. O sistema ótico tem a função de focalizar a energia proveniente da cena sobre o detetor. O sinal elétrico contínuo produzido pelo detetor é então processado e transformado, e a cada nível de radiância está associado um conjunto de coordenadas armazenadas em forma digital. O processo da transformação é chamado de **conversão analógica/digital (A/D)**. Em outras palavras, o sinal elétrico (analógico); que é produzido pelo detetor, é amostrado para armazenamento em forma digital. Esse processo é também chamado de **digitalização**.

Os pontos amostrados, descreve SONKA, HLAVAC & BOYLE (1993), são ordenados no plano bidimensional e estruturados usualmente em uma matriz. Um ponto amostrado é chamado de **pixel (picture element)**.

A magnitude de uma imagem amostrada é expressa como um valor digital **DN (digital number)** no processamento de imagens. A transição de valores contínuos da função-imagem a esses equivalentes digitais (DN) é denominado de **quantização**. Esses níveis de quantização devem no mínimo ser altos o suficiente para que a percepção humana possa detectar pequenos detalhes na imagem.

Geralmente os níveis de cinza que formam uma **imagem digital** são gravados em um byte de 8 bits, múltiplos de 2, gerando no caso 2^8 (256) níveis de cinza que podem variar de 0 a 255, e desta maneira a imagem pode ser analisada pelo computador. Então, um ponto infinitamente pequeno amostrado e representado no grid é chamado, como já mencionado anteriormente, de pixel na imagem digital, sendo esse uma unidade indivisível.

Os conceitos dados até aqui se aplicam a **imagens monocromáticas** que resultam em imagens de tons de cinza.

GREEN (1989) mostra na figura 2.2, uma concepção de um sensor que adquire imagens digitais da mesma cena em diferentes partes do espectro eletromagnético. A imagem de saída passa através de 3 filtros espectrais, sendo separadas em três componentes espectrais as quais são representadas na forma digital por matrizes. O conjunto das três matrizes resultantes define uma **imagem digital multispectral ou imagem digital colorida**.

Assim, para representar numericamente a imagem digital colorida é fornecido o nível de intensidade das três componentes: vermelho, verde e azul

que é associado a um valor numérico, onde o valor 0(zero) representa o nível de intensidade nulo e 255 ao de intensidade máxima. Já na imagem digital multiespectral, três ou mais componentes espectrais podem representar quaisquer informações da imagem. Pode-se citar a chamada **falsa-cor** em sensoriamento remoto que pode ser representada, por exemplo, pelas componentes verde, vermelho e infravermelho (CHUVIECO,1990).

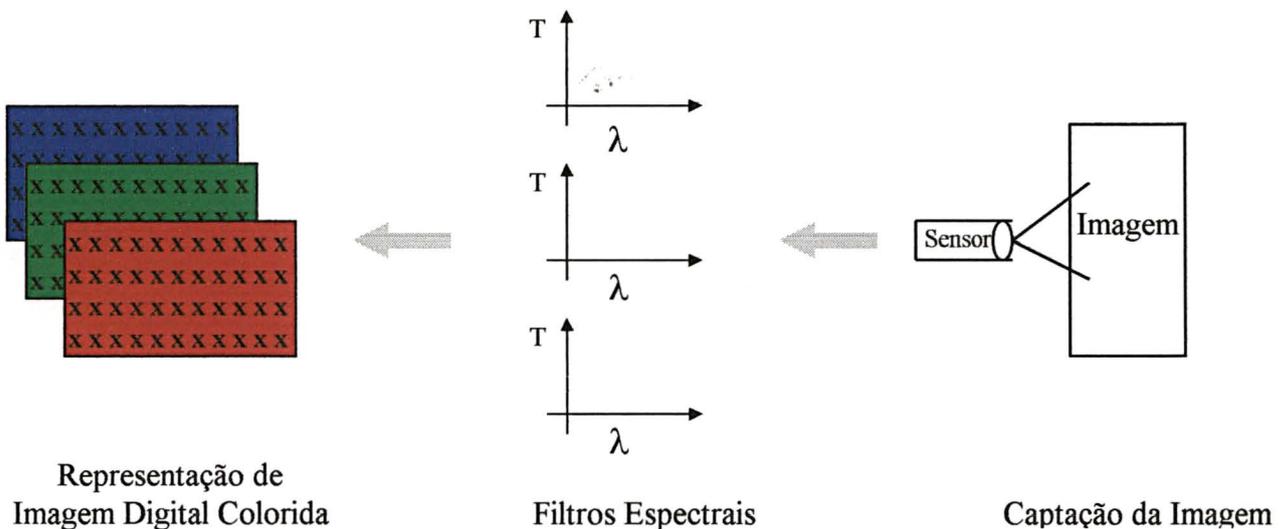


Figura 2.2. Concepção de Imagem Digital Colorida.

Fonte: reproduzida de GREEN (1989)

Existem vários sistemas de representação de cores, sendo os mais utilizados os sistemas **RGB** (Red-Green-Blue) e **IHS** (Intensity-Hue-Saturation).

O sistema **IHS** é usado como alternativa para descrever cores, onde **I**-intensidade é a soma das componentes coloridas (RGB), **H**-hue (tonalidade) é aproximadamente proporcional a média do comprimento de onda do espectro e **S**-Saturação é a deficiência de cor branca (SONKA, HLAVAC & BOYLE, 1993) ou ainda a pureza da cor em relação ao cinza (LILLESAND & KIEFER, 1994). Este sistema pode ser obtido através de uma transformação do sistema RGB e pode, também retornar a ele caso se deseje.

O sistema RGB, de acordo com THALMANN & THALMANN (1987), é derivado de um sistema definido em 1931 pela Comissão Internacional de L'Eclairage, conhecido como CIE. Este sistema define um

espaço baseado na combinação linear das três cores primárias, como já mencionado, vermelho(Red), verde (Green) e azul (Blue).

A representação do modelo, segundo BANON (1989) é feita através de um cubo unitário, num espaço referenciado por um sistema de coordenadas ortogonais, chamado de Cubo RGB, como mostra a figura 2.3.

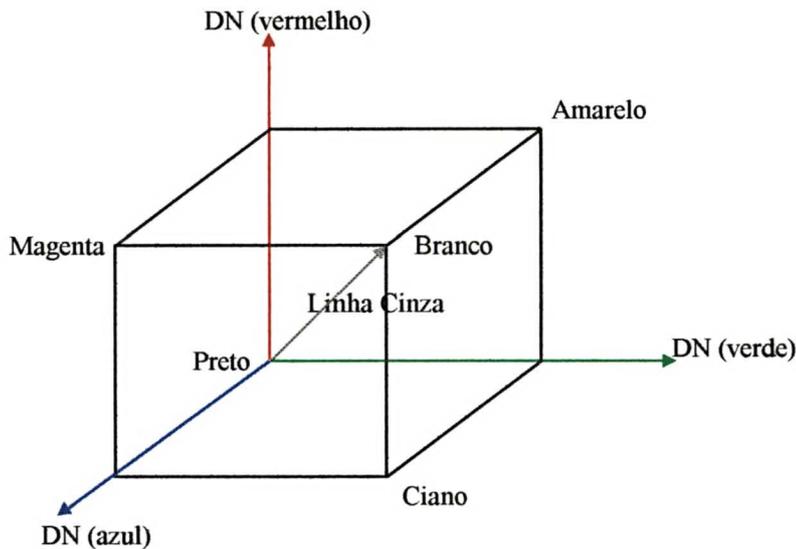


Figura 2.3. Cubo do RGB

Fonte: reproduzida de LILLESAND & KIEFER (1994)

O preto é representado na origem com coordenadas $(0,0,0)$, o branco $(1,1,1)$ e as três cores primárias são encontradas nos eixos principais. A cor vermelha tem coordenadas $(1,0,0)$, a verde $(0,1,0)$ e a azul $(0,0,1)$. As cores complementares ciano, magenta e amarelo estão localizadas nos vértices restantes com as coordenadas respectivas $(0,1,1)$, $(1,0,1)$ e $(1,1,0)$.

O sistema RGB se baseia no **processo aditivo** de composição de cores, onde qualquer cor pode ser obtida da soma das três cores primárias (vermelho, verde e azul), como pode-se observar na figura 2.4.

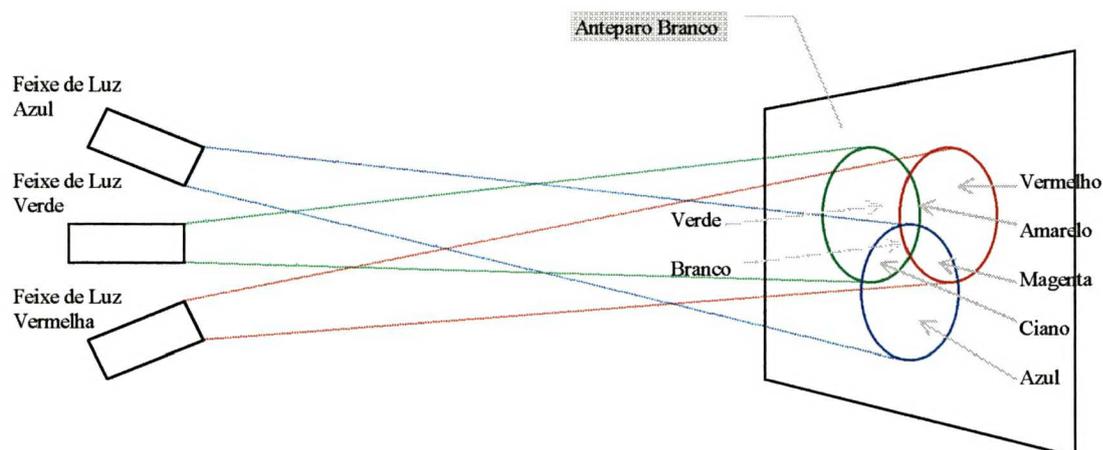


Figura 2.4. Processo Aditivo de Composição de Cores.

Fonte: reproduzida de NOVO (1992)

Após a imagem estar digitalizada, os dados podem sofrer modificações que são denominadas de técnicas de **processamento digital de imagens** que tem por objetivo fornecer ferramentas que possam facilitar a identificação e a extração de informações contidas em uma imagem que se deseja analisar. Pode-se citar como exemplo destas técnicas de processamento a Segmentação de Imagens cujo principal objetivo é dividir uma imagem em partes que tenham uma forte correlação com objetos de interesse contidos nela. Outra técnica poderia ser a de Detecção de Bordas de um objeto que se deseja analisar. Ainda, pode-se citar as técnicas de reconhecimento estatísticos de padrões em imagem que foram estudadas neste trabalho (SONKA, HLAVAC & BOYLE, 1995).

O objetivo das técnicas de classificação de imagem, de acordo com LILLESAND & KIEFER (1994), é categorizar automaticamente todos os pixels da imagem em classes ou temas. Normalmente, dados multiespectrais são usados para definir a classificação; isto é, o padrão espectral presente nos dados de cada pixel é usado como uma base numérica para classificação. Assim, o termo padrão não é referido somente a um caráter geométrico, mas sim a um conjunto de mensurações radiométricas obtidas de bandas de diferentes comprimentos de onda para cada pixel. O **reconhecimento de padrão espectral** refere-se a uma família de procedimentos de classificação que utiliza informações espectrais pixel-a-pixel como uma base de classificação automática.

O **Reconhecimento de padrões** (SONKA, HLAVAC & BOYLE 1993) baseia-se na atribuição de classes para estes objetos através do processo denominado de **Análise Discriminante**.

Análise Discriminante e Classificação são técnicas multivariadas interessadas com a separação de uma coleção de objetos distintos e que aloca novos objetos em grupos previamente definidos (JOHNSON & WICHERN, 1985).

Estas técnicas apesar de estarem claramente interligadas não devem ser confundidas. A **Análise Discriminante** se refere aos métodos de atribuição de classes a determinados conjuntos de dados. Como exemplo, a cada cor na imagem atribui-se uma classe, que vai ser diferenciada através dos métodos de Máxima Verossimilhança, Distância Mínima e de Fisher. No entanto, a **Classificação** se refere a alocação de um conjunto de pixels coloridos nas suas devidas classes correspondentes, feitas neste trabalho através de regras definidas para um dos métodos citados.

Discriminações podem ser feitas através de processos **supervisionados e não-supervisionados**. Os processos **supervisionados** são utilizados quando se conhece os padrões com os quais se deve analisar os dados, onde é tomada uma amostra de cada padrão e se define uma função discriminante para a mesma. Já os processos **não-supervisionados** não possuem um padrão conhecido, para tanto são criados padrões que separam os dados de acordo com suas características semelhantes.

Os métodos de classificação mais usados em Fotogrametria são: Máxima Verossimilhança, Distância Mínima, Paralelepípedo (supervisionados) e Clusters (não-supervisionados).

Os métodos propostos neste trabalho - Máxima Verossimilhança, Distância Mínima e de Fisher - são supervisionados e de concepção estatística.

O **Método de Máxima Verossimilhança** assume, segundo FOODY, McCULLOCH & YATES (1995), que os dados de cada classe tenham uma distribuição normal multivariada, como mostra a figura 2.5 apresentada por LILLESAND & KIEFER (1994), onde foram obtidas classes a partir de duas bandas espectrais (sensoriamento remoto) e plotados os valores de probabilidade em um gráfico tridimensional. A classificação, de acordo com MARDIA, KENT & BIBBY (1982), é feita a partir da distribuição normal de maior verossimilhança, descrita mais detalhadamente no item 3.1.1.

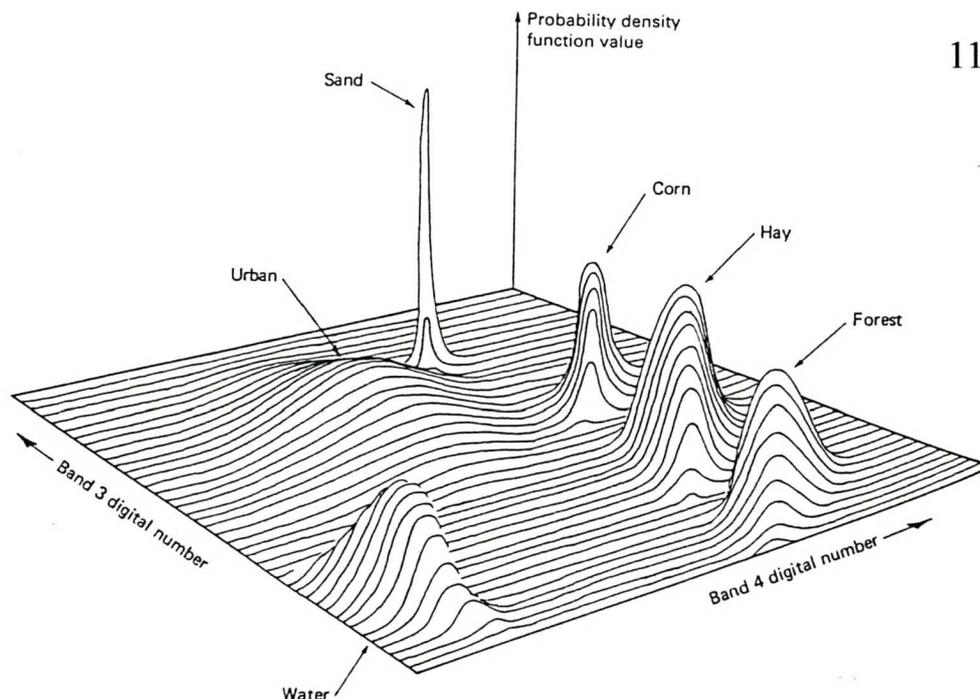


Figura 2.5: Funções Densidade de Probabilidade Definida por Máxima Verossimilhança . (LILLEDAND & KIEFER 1994)

A classificação pelo método de **Distância Mínima**, como mostra a para um caso bidimensional na figura 2.6, é obtida a partir da menor distância entre o pixel a ser classificado (x_0) e os valores médios obtidos das amostras treinadas (m_1 , m_2 e m_3).

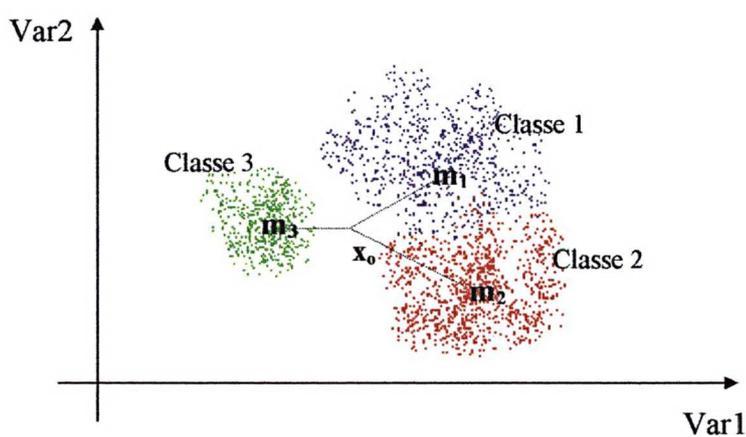


Figura 2.6 Representação do Método de Distância Mínima.

O **Método de Fisher**, apresentado no item 3.1.3, foi deduzido com o propósito de separar as populações (classes) tanto quanto possível, não sendo necessário assumir que as populações sejam normais multivariadas; no entanto deve-se assumir que suas matrizes de covariância sejam iguais. A idéia principal deste método é de transformar observações multivariadas em combinações lineares univariadas, com intuito de reduzir a dimensão de um grande número de características para relativamente poucas combinações lineares. Desta maneira, classifica-se um novo pixel em uma classe cuja combinação linear tenha menor distância quadrática da média das combinações lineares das classes observadas. Pode-se ver na figura 2.7 abaixo a representação gráfica do método para duas populações, onde \bar{x}_1 e \bar{x}_2 são respectivamente as médias das populações π_1 e π_2 e $y = \hat{l}'x$ é a combinação linear obtida para este caso bidimensional (JOHNSON & WICHERN, 1985).

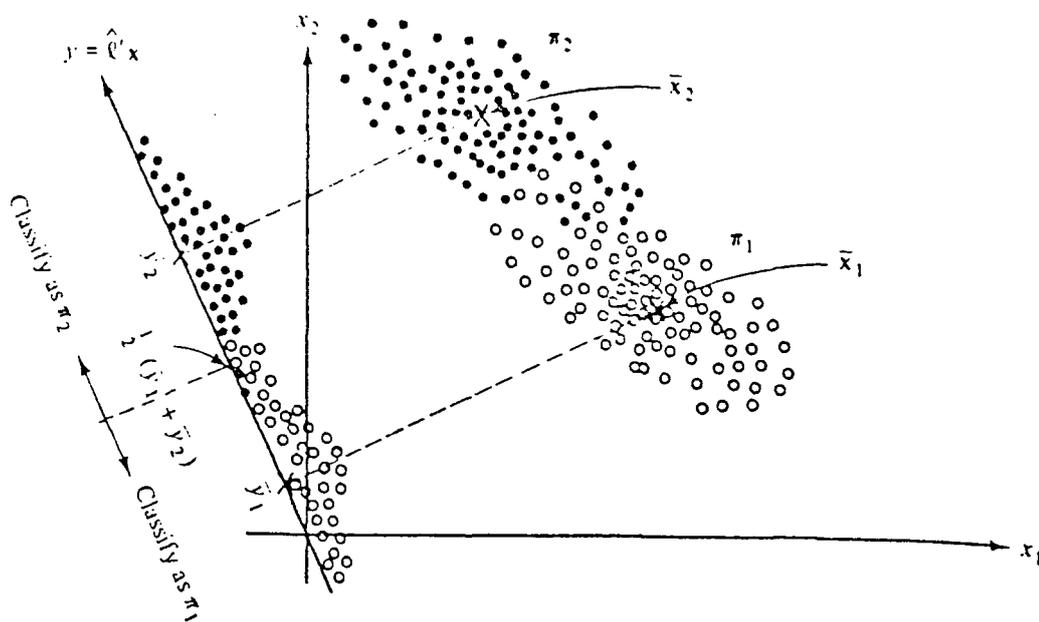


Figura 2.7: Representação do Procedimento de Fisher para Duas Populações.

Fonte: JOHNSON & WICHERN (1985)

Segundo YOUNG & FU (1986), a classificação é constituída de duas etapas, primeiramente se faz uma amostragem de dados para as várias classes de interesse, então se determina através de funções discriminantes os limites que as separam. Este processo é chamado de **treinamento** ou **aprendizado**. Para que se possa alocar estas amostra ou novos dados deve-se ter uma medida que separe uma classe da outra, chamada de **classificador**, que no caso são regras descritas pelas funções discriminante, como já mencionado anteriormente.

Para analisar o desempenho dos métodos utilizou-se o coeficiente *kappa*, deduzido por COHEN (1960) sendo este o mais usado para determinar a acurácia de classificações de imagens. Este é obtido de uma **Tabela de Contingência**, também chamada de **Matriz de Erro**, que compara o relacionamento entre os dados de referência e os dados alcançados pelo discriminante (LILLESAND & KIEFER, 1994).

O coeficiente Kappa possui distribuição aproximadamente normal e o teste de significância entre dois tipos de discriminantes podem ser obtidos através da estatística **Z** definida por COHEN (1960) e apresentada no item 3.2.1.

Utilizou-se também neste trabalho o **Teste de Proporções** que como o nome indica compara duas proporções e segundo DEVORE (1982), se comporta como uma distribuição normal padronizada **Z**, descrita no item 3.2.2.

No capítulo a seguir, ter-se-á uma descrição dos métodos de Máxima Verossimilhança, Distância Mínima e de Fisher; além dos testes estatísticos utilizados neste trabalho.

3. MÉTODOS E TESTES ESTATÍSTICOS:

A descrição dos métodos selecionados estão apresentados neste capítulo assim como os testes estatísticos necessários a comparação dos mesmos.

3.1. MÉTODOS ANALISADOS:

Os métodos selecionados neste trabalho tem concepção estatística e possuem algumas diferenças expostas a seguir.

O método de Máxima Verossimilhança considera que as classes sejam normais multivariadas e conseqüentemente depende do vetor médio e da matriz de covariância de cada classe, sendo assim classifica um novo valor na classe de maior verossimilhança.

O método de Distância Mínima depende somente do vetor médio e classifica segundo a classe de menor distância ao novo valor.

O método de Fisher transforma as observações multivariadas em combinações lineares univariadas, selecionando a classe cuja combinação linear possua menor distância à combinação linear de um novo valor. Ainda, este método considera que as classes são oriundas de uma população qualquer, não necessariamente normais multivariadas e depende do vetor médio e da matriz de covariância conjunta das classes.

A seguir os métodos discriminantes selecionados são apresentados incluindo as regras de classificação para um novo valor.

3.1.1. MÉTODO DE MÁXIMA VEROSSIMILHANÇA.

O desenvolvimento do método se baseia, segundo RICHARDS (1986), na Função Penalidade, que avalia as decisões incorretas.

Seja x um vetor de variáveis aleatórias pertencentes a uma classe π_k (no caso cores); $p(x/\pi_k)$ a probabilidade condicional de x ocorrer nessa classe (ou população) e $p(\pi_k)$ a probabilidade "a priori" da classe.

Então, $\lambda(i/k)$ é a Função Perda (MOIK, 1980), Função Penalidade (RICHARDS, 1986) ou também conhecida como Função Custo de Classificação Errada (JOHNSON & WICHERN, 1985) que designa a perda de alocação na classe π_i quando de fato $\lambda(i/k)$ pertence a π_k . Sendo, $\lambda(i/k)$ uma variável do tipo dicotômica, temos:

$$\lambda(i/k) = \begin{cases} 0 & , i = k \\ 1 & , i \neq k \end{cases} \quad i, k = 1, 2, \dots, m$$

Determinando-se a média de $\lambda(i/k)$ e considerando todas as possibilidades de π_k , obtem-se a **perda média condicional** apresentada abaixo para o caso multivariado, onde foi aplicado a esta variável aleatória discreta a definição de esperança matemática.

$$L_x(\pi_i) = \sum_{\substack{k=1 \\ i \neq k}}^m \lambda(i/k) \cdot p(\pi_k/x) \quad (3.1)$$

Esta função se refere a uma perda acumulada decorrente de valores observados de x que venham de algumas das classes avaliadas, de maneira que pode-se relacionar as funções de perda avaliadas de todas as classes para a classe π_i . Então, a regra de decisão que minimiza a perda média condicional é chamada de "Classificador Ótimo", "Classificador Bayesiano" (MOIKE, 1980) ou "Otimização de Bayes" (RICHARDS, 1986) dada por:

$$x \in \pi_i \quad \text{se } L(x, \pi_i) < L(x, \pi_k) \quad \forall \begin{matrix} k = 1, 2, \dots, m \\ i = 1, 2, \dots, m \\ k \neq i \end{matrix} \quad (3.2)$$

Geralmente, $p(\pi_k/x)$ é dita probabilidade "*a posteriori*" e pode-se determiná-la em termos das funções distribuição de probabilidade das classes $p(x/\pi_k)$, utilizando o Teorema de Bayes como segue:

$$p(\pi_k/x) = \frac{p(x/\pi_k) \cdot p(\pi_k)}{p(x)} \quad (3.3)$$

Aplicando o teorema acima na equação da perda média condicional tem-se:

$$L_x(\pi_i) = \frac{1}{p(x)} \sum_{k=1}^m \lambda(i/k) \cdot p(x/\pi_k) \cdot p(\pi_k) \quad (3.4)$$

fazendo:

$$l_x(\pi_i) = \sum_{k=1}^m \lambda(i/k) \cdot p(x/\pi_k) \cdot p(\pi_k) \quad (3.5)$$

fica-se com a expressão:

$$L_x(\pi_i) = \frac{1}{p(x)} \cdot l_x(\pi_i) \quad (3.6)$$

Dado que $p(x)$ é comum para todas as classes, a decisão pode ser tomada somente com base em $l_x(\pi_i)$.

Para este caso pode-se definir a função de perda como $\lambda(i/k) = 1 - \theta_{ik}$ com $\theta_{ii} = 1$ e θ_{ik} com $i \neq k$. Assim pode-se expressar:

$$l_x(\pi_i) = \sum_{k=1}^m p(x/\pi_k) \cdot p(\pi_k) - \sum_{k=1}^m \theta_{ik} \cdot p(x/\pi_k) \cdot p(\pi_k) \quad (3.7)$$

sabendo ser : $p(x) = \sum_{k=1}^m p(x/\pi_k) \cdot p(\pi_k)$, pelo Teorema da Probabilidade Total e

fazendo: $g_i(x) = \sum_{k=1}^m \theta_{ik} \cdot p(x/\pi_k) \cdot p(\pi_k)$ tem-se que:

$$l_x(\pi_i) = p(x) - g_i(x) \quad (3.8)$$

Novamente, pode-se retirar a probabilidade $p(x)$ comum para todas as classes não afetando na classificação, fica-se com a igualdade:

$$l_x(\pi_i) = -g_i(x) \quad (3.9)$$

Sabendo-se que $l_x(\pi_i)$ procede da expressão $L_x(\pi_i)$ e que representa a perda média condicional, assim o sinal negativo da expressão acima nos indica que quanto maior for $g_i(x)$ menor será a perda, então:

$$x \in \pi_i \text{ se } g_i(x) > g_j(x) \quad \forall j \neq i \quad (3.10)$$

É conveniente, segundo RICHARDS (1986), fazer $\theta_{ik} = \delta_{ik}$ que é a função delta de Kroneker definida por:

$$\delta_{ik} = \begin{cases} 1 & \text{para } i = k \\ 0 & \text{para } i \neq k \end{cases} \quad (3.11)$$

Nestas condições a expressão de $g_i(x)$ fica:

$$g_i(x) = p(x/\pi_i) \cdot p(\pi_i) \quad (3.12)$$

e a regra de decisão é:

$$x \in \pi_i \quad \text{se } p(x/\pi_i) \cdot p(\pi_i) > p(x/\pi_j) \cdot p(\pi_j) \quad \forall j \neq i \quad (3.13)$$

Esta regra, segundo RICHARDS (1986) é mais aceitável desde que $p(x/\pi_k)$ sejam conhecidos através do treinamento dos dados.

Assumindo que as distribuições de probabilidade de cada classe são da forma normal multivariada, tem-se que:

$$p(x/\pi_i) = (2\pi)^{(-n/2)} |\Sigma_i|^{-1/2} \cdot \exp\left\{-\frac{1}{2}(x - \mu_i)' \Sigma_i^{-1}(x - \mu_i)\right\} \quad (3.14)$$

Onde μ_i e Σ_i são respectivamente o vetor médio e a matriz de covariância dos dados da classe π_i que geralmente são desconhecidos, então a partir de uma amostra de tamanho n_i da população π_i pode-se obter as respectivas estimativas :

$$\bar{x}_i = \frac{1}{n_i} \sum_{r=1}^{n_i} x_{ir} \quad (3.15)$$

$$S_i = \sum_{i=1}^m \sum_{r=1}^{n_i} (x_{ir} - \bar{x}_i)(x_{ir} - \bar{x}_i)' \quad (3.16)$$

Ainda, sabendo ser o resultado $-\frac{n}{2} \ln(2\pi)$ comum para todos $g_i(x)$, tem-se a expressão:

$$g_i(x) = \ln p(\pi_i) - \frac{1}{2} \ln |S_i| - \frac{1}{2} (x - \bar{x}_i)' S_i^{-1} (x - \bar{x}_i) \quad (3.17)$$

Como $p(\pi_i)$ é suposta desconhecida, assume-se que todas as classes tenham igual probabilidade de ocorrer; pode-se então retirar esse fator, o que resultará em mais um fator comum a ser eliminado ($1/2$), pois estes não afetam na decisão para a classificação. Assim a Função Discriminante Amostral de Máxima Verossimilhança é dada por:

$$g_i(x) = -\ln|S_i| - (x - \bar{x}_i)' S_i^{-1} (x - \bar{x}_i) \quad (3.18)$$

E a regra de decisão fica:

$$x \in \pi_i \quad \text{se} \quad -\ln|S_i| - (x - \bar{x}_i)' S_i^{-1} (x - \bar{x}_i) > -\ln|S_j| - (x - \bar{x}_j)' S_j^{-1} (x - \bar{x}_j) \\ \forall j \neq i \quad i, j = 1, 2, \dots, m \quad (3.19)$$

Neste trabalho, as classes π_i designadas acima são as várias cores que se deseja discriminar e as observações x_i (vetores multivariados que correspondem as componentes RGB) são as amostras das respectivas cores, e conseqüentemente S_i serão as matrizes de covariância para cada tipo de amostra. Assim, quando se tem um vetor multivariado (RGB) x , que corresponder a um pixel na imagem, com o qual se deseja alocar em uma das classes ou cores, utiliza-se a regra acima.

3.1.2. MÉTODO DE DISTÂNCIA MÍNIMA.

O método de Distância Mínima classifica um pixel de uma classe em particular com base na classe de média mais próxima.

Este método é um caso particular do método de Máxima Verossimilhança descrito anteriormente, pois segundo RICHARDS (1986), este considera as matrizes de covariância de todas as classes como diagonais e iguais e ainda as variâncias de cada componente como idênticas ou relativamente próximas, assim tem-se que:

$$\Sigma_i = \sigma^2 I \quad \text{para o caso populacional} \quad (3.20)$$

$$S_i = s^2 I \quad \text{para o caso amostral} \quad (3.21)$$

onde: Σ_i = matriz de covariância populacional da classe i

S_i = matriz de covariância amostral da classe i

σ^2 = variância populacional

S^2 = variância amostral

I = matriz identidade

Então, na equação 3.18 dada por:

$$g_i(x) = -\ln|S_i| - (x - \bar{x}_i)' S_i^{-1} (x - \bar{x}_i)$$

onde \bar{x}_i = média amostral da classe i , pode-se substituir S_i por $s^2 I$ para todo i e tem-se:

$$g_i(x) = -\ln|s^2 I| - (x - \bar{x}_i)' s^2 I (x - \bar{x}_i) \quad (3.22)$$

Sendo $s^2 I$ constante para todas as classes e um fator que não discrimina, pode assim ser eliminada da função gerando o que se chama de distância quadrática que é dada pela expressão:

$$d(x, \bar{x}_i)^2 = (x - \bar{x}_i)' \cdot (x - \bar{x}_i) \quad (3.23)$$

Desta maneira a classificação pode ser feita com base na regra:

$$x \in \Pi_i \quad \text{se} \quad (x - \bar{x}_i)' (x - \bar{x}_i) < (x - \bar{x}_j)' (x - \bar{x}_j) \quad \forall, i \neq j \quad (3.24)$$

ou

$$\boxed{x \in \Pi_i \quad \text{se} \quad d(x, \bar{x}_i)^2 < d(x, \bar{x}_j)^2 \quad \forall, i \neq j} \quad (3.25)$$

Quando se deseja alocar um novo ponto, conclui-se que este deve ser classificado na classe que possua a menor distância quadrática até ele.

3.1.3. MÉTODO DE FISHER.

O Método Discriminante de Fisher para várias populações é uma extensão deste mesmo método para duas populações. Por esta razão, se faz necessário descrever, primeiramente, o método para duas populações, afim de melhorar o entendimento.

Sejam p variáveis aleatórias associadas a um vetor $X' = [X_1, X_2, \dots, X_p]$, onde as observações de valores de X difere de uma classe π_1 para outra π_2 . Considere-se os valores da primeira classe como sendo a população de x valores para π_1 e para a segunda classe com população de x valores para π_2 . Então, pode-se descrever funções densidades de probabilidade $f_1(x)$ e $f_2(x)$ para as duas populações.

A idéia de Fisher, segundo JOHNSON & WICHERN (1980), era a transformação de observações multivariadas x para observações univariadas y , e que uma das funções y derivadas das populações π_1 e π_2 sejam tão separáveis quanto possível.

Fisher sugeriu que os valores de y sejam combinações lineares, pois estas são funções de fácil manuseio matemático.

Definindo μ_{1y} a média da combinação linear de X pertencente a π_1 e μ_{2y} a π_2 , Fisher selecionou as combinações lineares que maximizam a distância (quadrática) entre μ_{1y} e μ_{2y} relativa a variabilidade dos valores de y .

Seja Y a combinação linear de X dado por:

$$Y = l'X \quad (3.26)$$

onde l' corresponde ao vetor de coeficientes.

Tem-se que:

$$\mu_{1y} = E(Y/\pi_1) = E(l'X/\pi_1) = l'E(X/\pi_1) \quad (3.27)$$

$$\mu_{2y} = E(Y/\pi_2) = E(l'X/\pi_2) = l'E(X/\pi_2) \quad (3.28)$$

fazendo:

$$E(X/\pi_1) = \mu_1 \quad (3.29)$$

e
$$E(X/\pi_2) = \mu_2 \quad (3.30)$$

tem-se:
$$\mu_{1y} = l'\mu_1 \quad (3.31)$$

$$\mu_{2y} = l' \mu_2 \quad (3.32)$$

A matriz de covariância considerada a mesma para as duas populações é dada por:

$$\sigma_y^2 = \text{Var}(l'X) = l' \text{Cov}(X) l = l' \Sigma l \quad (3.33)$$

onde

$$\Sigma = E(X - \mu_i)(X - \mu_i)', \quad i = 1, 2. \quad (3.34)$$

A melhor combinação linear é obtida da razão entre o quadrado da distância entre as médias e a variância de Y , assim:

$$\frac{(\mu_{1Y} - \mu_{2Y})^2}{\sigma_y^2} = \frac{(l' \mu_1 - l' \mu_2)^2}{l' \Sigma l} = \frac{l' (\mu_1 - \mu_2) (\mu_1 - \mu_2)' l}{l' \Sigma l} = \frac{(l' \delta)^2}{l' \Sigma l} \quad (3.35)$$

onde δ é a diferença dos valores médios, isto é:

$$\delta = (\mu_1 - \mu_2) \quad (3.36)$$

Note que a matriz $\delta \delta' = (\mu_1 - \mu_2) (\mu_1 - \mu_2)'$ contém os quadrados dos produtos cruzados dos diferentes componentes entre as médias das populações π_1 e π_2 .

Os coeficientes da combinação linear de Fisher $l = [l_1, l_2, \dots, l_p]$ são aqueles que maximizam a razão $\frac{(l' \delta)^2}{l' \Sigma l}$:

$$l = c \Sigma^{-1} \delta = c \Sigma^{-1} (\mu_1 - \mu_2) \quad \text{para } c \neq 0 \quad (3.37)$$

Escolhendo $c = 1$ obtêm-se:

$$Y = l' X = (\mu_1 - \mu_2)' \Sigma^{-1} X \quad (3.38)$$

que é conhecida como **Função Discriminante Linear de Fisher** e o máximo da razão é obtido por:

$$\text{MAX}_l \frac{(l \delta)^2}{l' \Sigma l} = \delta' \Sigma^{-1} \delta \quad (3.39)$$

Sendo a matriz de covariância uma matriz positiva definida, a prova da maximização da razão é feita aplicando a Lemma da Maximização, baseado na desigualdade de Cauchy-Schwarz (JOHNSON & WICHERN, 1980).

A Função Linear Discriminante converte as populações π_1 e π_2 que são multivariadas em populações univariadas de maneira que as médias das populações univariadas correspondentes são separadas tanto quanto possível.

A equação $Y = (\mu_1 - \mu_2)' \Sigma^{-1} X$ também pode ser empregada como um dispositivo de classificação (classificador).

Utilizando-se a equação 3.38 para uma nova observação X_0 , tem-se $y_0 = (\mu_1 - \mu_2)' \Sigma^{-1} X_0$ como o valor da função discriminante e ainda:

$$m = \frac{1}{2}(\mu_{1Y} + \mu_{2Y}) = \frac{1}{2}(l'\mu_1 + l'\mu_2) \quad (3.40)$$

ou

$$m = \frac{1}{2}(\mu_1 - \mu_2)' \Sigma^{-1} (\mu_1 + \mu_2) \quad (3.41)$$

que é o ponto médio entre as duas médias populacionais univariadas.

Assim pode-se mostrar que:

$$E(Y_0/\pi_1) - m \geq 0 \quad (3.42)$$

e

$$E(Y_0/\pi_2) - m < 0 \quad (3.43)$$

Desta maneira se X_0 for oriundo de π_1 , a esperança de Y_0 é maior ou igual ao ponto médio. Caso X_0 for oriundo de π_2 , a esperança de Y_0 é menor que o ponto médio. Então, a regra de classificação é dada por:

$$\text{alocar em } \pi_1 \text{ se } y_0 = (\mu_1 - \mu_2)' X_0 \geq m \quad (3.44)$$

ou

$$\text{alocar em } \pi_2 \text{ se } y_0 = (\mu_1 - \mu_2)' X_0 < m \quad (3.45)$$

Sabendo-se que as quantidades populacionais são raramente conhecidas, a regra acima não pode ser implementada a não ser que l e m possam ser estimados por observações que já tenham sido corretamente classificadas.

Suponha que se tenha n_1 observações da variável aleatória multivariada $X' = [X_1, X_2, \dots, X_p]$ de π_1 e n_2 observações de π_2 . Os vetores de dados respectivos a π_1 e π_2 são:

$$X_1 = [X_{11}, X_{12}, \dots, X_{1n_1}] \quad (3.46)$$

$$X_2 = [X_{21}, X_{22}, \dots, X_{2n_2}] \quad (3.47)$$

Para estes dados, os vetores amostrais de média e as matrizes amostrais de covariância são dadas por:

$$\bar{x}_1 = \frac{1}{n_1} \sum_{j=1}^{n_1} x_{1j} \quad (3.48)$$

$$S_1 = \frac{1}{n_1 - 1} \sum_{j=1}^{n_1} (x_{1j} - \bar{x}_1)(x_{1j} - \bar{x}_1)' \quad (3.49)$$

$$\bar{x}_2 = \frac{1}{n_2} \sum_{j=1}^{n_2} x_{2j} \quad (3.50)$$

$$S_2 = \frac{1}{n_2 - 1} \sum_{j=1}^{n_2} (x_{2j} - \bar{x}_2)(x_{2j} - \bar{x}_2)' \quad (3.51)$$

Como foi assumido que as populações tem a mesma matriz de covariância Σ , as matrizes de covariância S_1 e S_2 podem então ser combinadas para se obter uma única estimativa não tendenciosa de Σ , dada pela matriz de covariância conjunta:

$$S_c = \frac{(n_1 - 1)S_1 + (n_2 - 1)S_2}{(n_1 + n_2 - 2)} \quad (3.52)$$

Substituindo as quantidades μ_1 , μ_2 e Σ por respectivamente \bar{x}_1 , \bar{x}_2 e S_c , tem-se que a Função Discriminante Linear Amostral de Fisher para duas populações é dada por:

$$y = \hat{l}'x = (\bar{x}_1 - \bar{x}_2)' S_c^{-1} x \quad (3.53)$$

A estimativa do ponto médio entre duas médias amostrais univariadas, $\bar{y}_1 = \hat{l}'\bar{x}_1$ e $\bar{y}_2 = \hat{l}'\bar{x}_2$ é dado por:

$$\hat{m} = \frac{1}{2}(\bar{y}_1 - \bar{y}_2) = \frac{1}{2}(\bar{x}_1 - \bar{x}_2)' S^{-1}(\bar{x}_1 + \bar{x}_2) \quad (3.54)$$

E a regra de classificação mostra que se deve:

- alocar x_0 em π_1 se:

$$y_0 = (\bar{x}_1 - \bar{x}_2)' S_c^{-1} x_0 \geq \hat{m} \quad (3.55)$$

ou

$$y_0 - \hat{m} \geq 0 \quad (3.56)$$

- alocar x_0 em π_2 se:

$$y_0 - \hat{m} < 0 \quad (3.57)$$

Esta regra pode ser estendida para várias populações que não necessariamente tenham a mesma distribuição de probabilidade, entretanto necessita-se da suposição de que as matrizes das diversas populações sejam iguais; isto é, $\Sigma_1 = \Sigma_2 = \dots = \Sigma_g = \Sigma$.

Seja $\bar{\mu}$ o vetor médio combinado dos grupos e B_0 a soma dos produtos cruzados "entre" os grupos populacionais, tais que:

$$\bar{\mu} = \frac{1}{g} \sum_{i=1}^g \mu_i \quad (3.58)$$

$$B_0 = \sum_{i=1}^g (\mu_i - \bar{\mu})(\mu_i - \bar{\mu})' \quad (3.59)$$

onde se considera a combinação linear $Y = l'X$ e obtêm-se o valor esperado que se segue:

$$E(Y) = l'E(X/\pi_i) = l'\mu_i \quad (3.60)$$

para a população π_i e a variância:

$$ar(Y) = l'Cov(X)l = l'\Sigma l \quad (3.61)$$

para todas as populações.

Então, a média global para todas as populações é dada por:

$$\bar{\mu}_y = \frac{1}{g} \sum_{i=1}^g \mu_{iY} = \frac{1}{g} \sum_{i=1}^g l' \mu_i = l' \bar{\mu} \quad (3.62)$$

e a razão entre a soma dos quadrados das distâncias de todas as populações para a média global de Y e a variância de Y é dada por:

$$\frac{\sum_{i=1}^g (\mu_{iY} - \bar{\mu}_y)^2}{\sigma_Y^2} = \frac{\sum_{i=1}^g (l' \mu_i - l' \bar{\mu})^2}{l' \Sigma l} = \frac{l' \left(\sum_{i=1}^g (\mu_i - \bar{\mu})(\mu_i - \bar{\mu})' \right) l}{l' \Sigma l} = \frac{l' B_0 l}{l' \Sigma l} \quad (3.63)$$

Essa razão mede a variabilidade "entre" os grupos de valores relativos Y com a variabilidade comum "dentro" dos grupos.

Analogamente para o caso de 2 populações, pode-se selecionar l para maximizar a razão acima. Para tanto é conveniente normalizar l de tal maneira que $l' \Sigma l = 1$.

Seja $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_s > 0$, tal que $s \leq \min(g-1, p)$ autovalores não nulos de $\Sigma^{-1} B_0$ e e_1, e_2, \dots, e_s os autovetores correspondentes normalizados, tal que $e' \Sigma e = 1$. Então o vetor dos coeficientes l que maximiza a razão:

$$\frac{l' B_0 l}{l' \Sigma l} = \frac{l' \left[\sum_{i=1}^g (\mu_i - \bar{\mu})(\mu_i - \bar{\mu})' \right] l}{l' \Sigma l} \quad (3.64)$$

é dado por $l_1 = e_1$. A combinação linear $l_1' X$ é chamada de 1º discriminante. O valor $l_2 = e_2$ maximiza a razão anterior, sujeito a covariância $Cov(l_1' X, l_2' X) = 0$ e a combinação linear $l_2' X$ é chamada de 2º discriminante. Continuando, tem-se que $l_k = e_k$ maximiza a razão sujeita a $Cov(l_k' X, l_i' X) = 0$, para todo $i < k$. Aqui $l_k' X$ é chamado de k-ésimo determinante e para todos os discriminante a variância é igual a unidade.

A prova desse resultado é dado pelo Teorema da Decomposição Espectral, onde temos que $\Sigma = P' \Lambda P$ onde P é o vetor de autovalores e Λ é uma matriz diagonal com elementos positivos, λ_1 .

Sabendo que :

$$\Sigma^{1/2} = \sum_{i=1}^k \sqrt{\lambda_i} \cdot e_i \cdot e_i' = P \Lambda^{1/2} P' \quad (3.65)$$

é uma matriz raiz quadrada simétrica e que sua inversa é dada por:

$$\Sigma^{-1/2} = P' \Lambda^{-1/2} P \quad (3.66)$$

satisfaz as propriedades,

$$1. \Sigma^{1/2} \Sigma^{1/2} = \Sigma \quad (3.67)$$

$$2. \Sigma^{1/2} \Sigma^{-1/2} = I = \Sigma^{-1/2} \Sigma^{1/2} \quad (3.68)$$

$$3. \Sigma^{-1/2} \Sigma^{-1/2} = \Sigma^{-1} \quad (3.69)$$

Fazendo $a = \Sigma^{1/2} l$ então:

$$a'a = l' \Sigma^{1/2} \Sigma^{1/2} l = l' \Sigma l \quad (3.70)$$

e

$$a' \Sigma^{-1/2} B_0 \Sigma^{-1/2} a = l' \Sigma^{1/2} \Sigma^{-1/2} B_0 \Sigma^{-1/2} \Sigma^{1/2} l = l' B_0 l \quad (3.71)$$

Consequentemente, o problema se reduz a maximização expressa em a :

$$\frac{a' \Sigma^{-1/2} B_0 \Sigma^{-1/2} a}{a'a} \quad (3.72)$$

Pelo teorema de maximização de formas quadráticas para pontos em uma unidade esférica, tem-se o máximo da razão em λ_1 que é o maior autovalor de $\Sigma^{-1/2} B_0 \Sigma^{-1/2}$. Este máximo ocorre quando $a = e_1$, o autovetor normalizado associado com λ_1 . Porque se:

$$e_1 = a = \Sigma^{1/2} l_1 \Rightarrow l_1 = \Sigma^{-1/2} e_1 \quad (3.73)$$

$$Var(l_1' X) = l_1' \Sigma l_1 = e_1' \Sigma^{-1/2} \Sigma \Sigma^{-1/2} e_1 = e_1' \Sigma^{-1/2} \Sigma^{1/2} \Sigma^{1/2} \Sigma^{-1/2} e_1 = e_1' e_1 = 1 \quad (3.74)$$

Sabendo-se ser a perpendicular a e_1 e que maximiza a razão $\frac{a' \Sigma^{-1/2} B_0 \Sigma^{-1/2} a}{a'a}$. Então quando $a = e_2$ este será o autovetor normalizado correspondente para λ_2 . Assim:

$$l_2 = \Sigma^{-1/2} e_2 \quad (3.75)$$

$$\text{Cov}(l'_2 X, l'_1 X) = l'_2 \Sigma l_1 = e'_2 \Sigma^{1/2} \Sigma \Sigma^{1/2} e_1 = e'_2 e_1 = 0 \quad (3.76)$$

desde que $e_2 \perp e_1$.

Similarmente teremos $\text{ar}(l'_2) = l'_2 \Sigma l_2 = e'_2 e_2 = 1$ e continuando neste mesmo raciocínio temos que $a = e_k$ maximiza a razão $\frac{a' \Sigma^{-1/2} B_0 \Sigma^{-1/2} a}{a' a}$ sendo $a \perp e_1, e_2, \dots, e_{k-1}$ e $l_k = \Sigma^{-1/2} e_k$ satisfaz:

$$\text{Cov}(l'_k X, l'_i X) = l'_k \Sigma l_i = e'_k \Sigma^{-1/2} \Sigma \Sigma^{-1/2} e_i = e'_k e_i = \begin{cases} 0 & \text{se } i < k \\ 1 & \text{se } i = k \end{cases} \quad (3.77)$$

Então, se λ e e são respectivamente autovalores e autovetores de $\Sigma^{-1/2} B_0 \Sigma^{-1/2}$, tem-se:

$$\Sigma^{-1/2} B_0 \Sigma^{-1/2} e = \lambda e \quad (3.78)$$

e multiplicando os dois termos por $\Sigma^{-1/2}$ fica:

$$\Sigma^{-1/2} \Sigma^{-1/2} B_0 \Sigma^{-1/2} e = \lambda \Sigma^{-1/2} e \quad (3.79)$$

ou

$$\Sigma^{-1} B_0 (\Sigma^{-1/2} e) = \lambda (\Sigma^{-1/2} e) \quad (3.80)$$

Então, $\Sigma^{-1} B_0$ tem os mesmos autovalores que $\Sigma^{-1/2} B_0 \Sigma^{-1/2}$, mas os correspondentes autovetores são proporcionais a $\Sigma^{-1/2} e = l$, como se afirmou.

Assim, $l' \Sigma l = e' \Sigma^{-1/2} \Sigma \Sigma^{-1/2} e = e' e = 1$ e as combinações lineares $l'_1 X, l'_2 X, \dots, l'_s X$, correspondem a autovalores não nulos de $\Sigma^{-1} B_0$, formando uma série de discriminantes com :

$$\text{Var}(l'_i X) = 1 \quad (3.81)$$

$$\text{Cov}(l'_i X, l'_k X) = 0 \quad \text{para } i \neq k \quad (3.82)$$

Sabendo que μ_i e Σ são geralmente desconhecidos, pode-se então treinar uma série de observações corretamente classificadas. Supondo ser a série treinada uma amostra aleatória de tamanho n_i da população π_i , onde $i = 1, 2, \dots, g$, tem-se o vetor de média amostral é dado por:

$$\bar{x}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} x_{ij} \quad (3.83)$$

e o vetor de média global é definido abaixo, oriundo do vetor médio obtido de todas as observações de uma série treinada.

$$\bar{x} = \frac{\sum_{i=1}^g n_i \bar{x}_i}{\sum_{i=1}^g n_i} = \frac{\sum_{i=1}^g \sum_{j=1}^{n_i} x_{ij}}{\sum_{i=1}^g n_i} \quad (3.84)$$

A matriz B_0 que corresponde a matriz soma dos produtos cruzados "entre" os grupos populacionais pode ser estimada por:

$$\hat{B}_0 = \sum_{i=1}^g (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})' \quad (3.85)$$

A estimativa de Σ se baseia na matriz de soma dos produtos cruzados "dentro" dos grupos amostrais apresentada abaixo.

$$W = \sum_{i=1}^g (n_i - 1) S_i = \sum_{i=1}^g \sum_{j=1}^{n_i} (x_{ij} - \bar{x}_i)(x_{ij} - \bar{x}_i)' \quad (3.86)$$

onde S_i é a matriz de covariância de cada grupo.

Então a estimativa de Σ chamada de matriz de covariância conjunta é dada por:

$$\hat{\Sigma} = S_c = \frac{W}{(n_1 + n_2 + \dots + n_g - g)} \quad (3.87)$$

Tendo em vista que o mesmo \hat{l} que maximiza $\frac{\hat{l}' \hat{B}_0 \hat{l}}{\hat{l}' S_c \hat{l}}$ também maximiza $\frac{\hat{l}' \hat{B}_0 \hat{l}}{\hat{l}' W \hat{l}}$, então pode-se apresentar a otimização de \hat{l} na forma de autovetores, \hat{e}_i de $W^{-1} \hat{B}_0$ porque se $W^{-1} \hat{B}_0 \hat{e} = \hat{\lambda} \hat{e}$ então:

$$S_c^{-1} \hat{B}_0 \hat{e} = \hat{\lambda} (n_1 + n_2 + \dots + n_g - g) \hat{e} \quad (3.88)$$

Portanto o **Discriminante Amostral de Fisher** para várias populações pode ser definido a seguir.

Seja $\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_s > 0$ os autovalores não nulos de $W^{-1}\hat{B}_0$ e $\hat{e}_1, \hat{e}_2, \dots, \hat{e}_s$ os correspondentes autovetores.

Então, o vetor de coeficientes \hat{l} que maximiza a razão $\frac{\hat{l}'\hat{B}_0\hat{l}}{\hat{l}'W\hat{l}}$ é dado por $\hat{l}_1 = \hat{e}_1$ e a combinação linear $\hat{l}_1'x = \hat{e}_1'x$ é chamada de 1º Discriminante Amostral; $\hat{l}_2 = \hat{e}_2$ produz o 2º Discriminante Amostral $\hat{l}_2'x = \hat{e}_2'x$, generalizando tem-se que $\hat{l}_k'x = \hat{e}_k'x$ é o k-ésimo Discriminante Amostral para $k \leq s$.

O resultado fornece subsídios para classificar novos valores, dado que se determine as distâncias euclidianas entre o ponto selecionado e os discriminantes; e que a população escolhida para este seja o de mínima distância.

Para tanto, segue a regra:

Alocar x na população π_k se:

$$\sum_{j=1}^r (\hat{y}_j - \bar{y}_{kj})^2 = \sum_{j=1}^r [\hat{l}_j'(x - \bar{x}_k)]^2 \leq \sum_{j=1}^r [\hat{l}_j'(x - \bar{x}_i)]^2 \quad \text{para todo } i \neq k \quad (3.89)$$

onde:

$$\hat{y}_j = \hat{l}_j'x \quad (3.90)$$

e

$$\bar{y}_{kj} = \hat{l}_j'\bar{x}_k \quad \text{com } r \leq s \quad (3.91)$$

Dado que, as populações π_k mencionadas aqui se referem, no propósito deste trabalho, as cores à serem discriminadas, assim x e \bar{x}_k respectivamente o vetor que corresponderia ao pixel analisado e a média amostral da amostra tomada para determinar cada população, isto é, a cor.

Desta feita aloca-se um novo pixel à classe (cor) que contenha a menor distância quadrática entre as combinações lineares do ponto com um valor médio obtido pelos dados treinados.

3.2. TESTES ESTATÍSTICOS.

Quando se deseja testar hipóteses, de acordo com (COSTA NETO, 1977), denota-se por H_0 a hipótese a ser testada e por H_1 a hipótese alternativa, assim considera-se H_1 como hipótese complementar de H_0 . O teste estatístico leva a aceitação ou rejeição da hipótese H_0 , onde H_0 sempre indica que não há diferença entre duas variáveis.

O que leva a possibilidade de cometer dois tipos de erros, que são:

erro tipo I - rejeitar H_0 , sendo ela verdadeira.

erro tipo II - aceitar H_0 , sendo ela falsa.

As probabilidades desses dois tipos de erros são denotadas por α e β , respectivamente, onde a probabilidade de cometer o erro tipo I é denominada também de nível de significância do teste. Isto significa dizer que, ao se realizar testes, se está sujeito a um risco α de cometer o erro tipo I. Assim, a decisão de aceitar ou rejeitar a hipótese testada H_0 pode depender do nível de significância adotado, isto é, um resultado experimental obtido pode ser ou não significativo, dependendo do α adotado, sendo os valores mais utilizados na prática de $\alpha = 1\%$ e $\alpha = 5\%$.

Tendo fixado o valor de α , procura-se uma distribuição que se aproxima dos dados a serem testados, para que se possa determinar a estatística do teste. Estabelece-se a seguir os valores críticos da estatística, que são encontrados em tabelas a partir do nível de significância estabelecido. Estes valores determinam um intervalo de tal modo que se o valor encontrado na estatística do teste cair fora dele, isto indica que se deve rejeitar a hipótese H_0 .

Para este trabalho necessitou-se de dois tipos de testes, descritos nos itens 3.2.1 e 3.2.2. O primeiro se refere a um teste que compara o desempenho de dois métodos, onde se fixou o nível de significância de 5%. No segundo caso o teste é feito para comparações entre proporções de erros de classificação entre os diversos tipos de amostras propostas no capítulo 5.

3.2.1 AVALIAÇÃO DO DESEMPENHO DE CLASSIFICAÇÃO.

O estabelecimento da performance dos métodos é um fator de suma importância neste trabalho, dado que o seu objetivo maior é comparar os métodos de Máxima Verossimilhança, Distância Mínima e de Fisher.

Segundo MASCARENHAS & VELASCO (1984), a técnica mais comumente usada é retirar uma parte da imagem(amostra) para treinamento e outra para testar o discriminante.

Neste trabalho, optou-se por utilizar uma imagem gerada no *software* (MATLAB) para que se pudesse ter total controle sobre a mesma e criou-se amostras de vários tipos para que se observasse o desempenho dos métodos frente a qualquer tipo de amostra que possa ocorrer na prática.

Separou-se os dados em dois tipos: classificados e de referência (controle). Os dados chamados classificados são resultantes da aplicação

dos métodos propostos e os de referência são originários de uma imagem gerada com total controle.

Esta técnica foi utilizada para todos os experimentos descritos no capítulo 7, exceto no experimento 6 onde se captou uma imagem real digitalizada. Neste caso específico selecionou-se uma amostra de 4 elementos de cada cor para treinamento e após a classificação da imagem esta foi comparada com os dados de referência obtidos de uma classificação visual pixel a pixel.

Para que se possa comparar os métodos, LILLESANT & KIEFER (1994), mostrou-se que um dos meios mais comuns de expressar a acurácia é o cálculo de uma tabela denominada de **Tabela de Contingência** ou matriz de **Matriz Erro**. Esta tabela ou matriz, verifica o relacionamento entre os dados de referência, previamente conhecidos, com os dados obtidos pelo classificador, onde as linhas dessa matriz representam os dados de referência e as colunas os dados treinados e classificados pelo método escolhido.

Para que se possa entender melhor esta tabela, selecionou-se, como exemplo, uma imagem composta de três cores distintas (verde, vermelho e azul) e o método de Máxima Verossimilhança, cujo resultado aparece na tabela 3.1 que segue:

Tabela 3.1. Tabela de Contingência do Método de Máxima Verossimilhança com Amostras de Médias Próximas.

DADOS CLASSIFICADOS					
	Cor	Verde	Vermelho	Azul	total
DADOS	Verde	0	0	25	25
DE	Vermelho	0	0	25	25
REFERÊNCIA	Azul	0	0	25	25
	total	0	0	75	75

Fonte: Dados obtidos no presente trabalho no experimento 1 do 1^o caso.

Na matriz de erro, obtida da tabela acima, as colunas se referem ao número de pixels classificados pelo método e as linhas indicam os dados de referência para cada cor, com isto os valores apresentados na diagonal principal mostra a quantidade de dados corretamente classificados.

$$TCM = \begin{bmatrix} 0 & 0 & 25 \\ 0 & 0 & 25 \\ 0 & 0 & 25 \end{bmatrix} \quad \begin{array}{l} TCM = \text{Matriz Erro do Método} \\ \text{Máxima Verossimilhança} \end{array}$$

Desta maneira a tabela ou a matriz permitem retirar várias conclusões sobre a performance, tais como o total de classificação correta para cada cor observado na diagonal principal da matriz. Pode-se ainda verificar que fora da diagonal obtem-se os erros de exclusão e inclusão como mostra a primeira linha da tabela 3.1 onde a cor verde não foi classificada corretamente, sendo considerada como azul pelo classificador, caracterizando um erro de exclusão da cor verde e erro de inclusão na cor azul, o mesmo acontecendo para a cor vermelha na segunda linha.

Assim, na diagonal principal verifica-se que somente a cor azul foi classificada corretamente.

Somando-se os elementos da diagonal principal, que como já foi dito, corresponde ao total de classificação correta, e dividindo pelo número total de pixels observados, ter-se-á um valor denominado de **acurácia total**, que pode auxiliar numa primeira instância na caracterização da performance do classificador. Assim, temos:

$$OA = \frac{\sum_{i=1}^r x_{ii}}{N} \quad (3.92)$$

onde: $OA = \text{overall accuracy}$ (acurácia total)

x_{ii} = número de observações na diagonal principal

N = número total de observações

No caso do exemplo dado teremos:

$$OA = \frac{0+0+25}{75} = 0,3333 = 33,33\%$$

Para que se possa fazer uma comparação desta tabela de contingência com uma outra em que fosse utilizado outro método, recorre-se a uma estatística κ denominada de *kappa*, que mede a compatibilidade de uma tabela de contingência; ou seja, mede a compatibilidade dos dados classificados com os dados de referência (ZHUANG, ENGEL, XIONG & JOHANNSEN, 1995). A estatística κ é calculada utilizando as seguintes proporções:

P_o = proporção de concordância (obtida na diagonal).

P_e = proporção esperada de concordância (obtida no total das marginais).

Então kappa é dado por:

$$\kappa = \frac{P_o - P_e}{1 - P_e} \quad (3.93)$$

onde:

$$P_o = \frac{\sum_{i=1}^m x_{ii}}{N} \quad (3.94)$$

$$P_e = \frac{\sum_{i=1}^m x_{i+} \cdot x_{+i}}{N^2} \quad (3.95)$$

onde:

m = número de linhas na tabela de contingência.

x_{ii} = número de observações na linha i e coluna i .

x_{i+} = total de observações na linha i .

x_{+i} = total de observações na coluna i .

N = número total de observações.

Assim para o exemplo ter-se-ia:

$$P_o = OA = 0,3333$$

$$P_e = \frac{(25 \times 0) + (25 \times 0) + (25 \times 75)}{75^2} = 0,3333$$

$$\kappa = \frac{0,3333 - 0,3333}{1 - 0,3333} = 0$$

A comparação entre os métodos pode ser feita utilizando-se somente a medida de acurácia total, porém optou-se pela estatística *Kappa* por esta incorporar os elementos que não estão na diagonal principal da matriz com um produto de linhas e colunas, enquanto que a acurácia total inclui somente os dados classificados corretamente.

Uma das principais vantagens da estatística *Kappa* é a habilidade para usar este valor como uma base para determinar a significância estatística de alguma tabela dada ou a diferença entre tabelas, como por

exemplo, a performance individual de classificação utilizando diferentes métodos discriminantes, como é o caso. O teste é baseado em uma variância estimada de *Kappa* e usando um teste *Z* para determinar se há diferença significativa entre duas tabelas.

Então, segundo COHEN (1960), uma aproximação da variância do coeficiente verificada assintoticamente quando *N* é grande (≥ 30), é dada por:

$$\hat{\sigma}_k^2 = \frac{P_0(1 - P_0)}{N(1 - P_c)^2} \quad (3.96)$$

Sabendo que a diferença entre duas estatísticas *kappa* quando *N* é suficientemente grande tem uma distribuição aproximadamente normal, o teste de significância da diferença entre dois coeficientes *k* independentes pode ser feito por:

$$Z = \frac{\kappa_i - \kappa_j}{\sqrt{\hat{\sigma}_{ki}^2 + \hat{\sigma}_{kj}^2}} \quad \text{com } i \neq j \text{ e } i, j \text{ variando de 1 até 3} \quad (3.97)$$

onde:

k_1 = coeficiente Kappa para o Método de Máxima Verossimilhança

k_2 = coeficiente Kappa para o Método de Fisher

k_3 = coeficiente Kappa para o Método de Distância Mínima.

σ_{k1}^2 = variância amostral do coeficiente Kappa da tabela de contingência onde foi utilizado o Método de Máxima Verossimilhança para classificação.

σ_{k2}^2 = variância amostral do coeficiente Kappa da tabela de contingência onde foi utilizado o Método de Fisher para classificação.

σ_{k3}^2 = variância amostral do coeficiente Kappa da tabela de contingência onde foi utilizado o Método de Distância Mínima para classificação.

Ao comparar o método de Máxima Verossimilhança com o de Fisher para o exemplo exposto acima vemos que:

$$\kappa_1 = 0$$

$$\kappa_2 = 1$$

$$\sigma_{k1}^2 = 0,0067$$

$$\sigma_{k2}^2 = 0$$

$$Z_{cfm} = \frac{0 - 1}{\sqrt{0,0067 + 0}} = -12,2169$$

Sabendo ser as hipóteses estatísticas:

H_0 = Não há diferença entre as performances dos dois métodos

H_1 = Há diferença entre as performances dos dois métodos.

e observando o valor de Z padronizado ao nível de significância (probabilidade de rejeitar H_0 dado que ela é verdadeira - erro tipo I) de 5%, que corresponderia a um valor de $\pm 1,96$, pode-se ver que o valor de Z_{cfm}

(-12,2169), correspondente a estatística Z calculada pelo teste e que compara os dois métodos, se encontra fora do intervalo proposto (fora da região de aceitação de H_0), o que indica uma rejeição da hipótese H_0 e donde conclui-se que os dois métodos possuem performance diferentes ao discriminar as cores verde, vermelho e azul.

Comparou-se através deste teste os métodos dois a dois, isto é, testou-se o método de Máxima Verossimilhança para verificar se diferenciava, quanto a performance, com o de Fisher e o de Distância Mínima separadamente; e também se havia diferença significativa entre os métodos de Fisher e de Distância Mínima; como se observa em alguns resultados do capítulo 5.

3.2.2 TESTE PARA DUAS PROPORÇÕES.

Para testar hipóteses referentes às diferenças entre duas proporções populacionais, tais como descritas abaixo, e sabendo-se que as amostras tem tamanho " n_1 " para a primeira população e " n_2 " para a segunda, deve-se assumir que as variáveis " X_1 " e " X_2 " que poderiam representar o número de indivíduos em cada amostra passariam a caracterizar as proporções " p_1 " e " p_2 ". Assim, " X_1 " seria uma distribuição aproximadamente binomial com parâmetros " n_1 " e " p_1 " e " X_2 " também seria binomial com parâmetros " n_2 " e " p_2 ".

$H_0: p_1 - p_2 = 0$ (Hipótese Nula: Não há diferença entre as proporções)

$H_a: p_1 - p_2 \neq 0$ (Hipótese Alternativa: há diferença entre as proporções)

Assumindo que " X_1 " e " X_2 " são variáveis aleatórias independentes, tem-se que, segundo DEVORE (1982), a estatística do teste, dada abaixo,

deverá ser uma distribuição normalmente distribuída (Z) com média $(\hat{p}_1 - \hat{p}_2)$ e variância igual a $(\frac{\hat{p}_1 \cdot \hat{q}_1}{n_1} + \frac{\hat{p}_2 \cdot \hat{q}_2}{n_2})$.

$$Z = \frac{(\hat{p}_1 - \hat{p}_2) - 0}{\sqrt{\frac{\hat{p}_1 \cdot \hat{q}_1}{n_1} + \frac{\hat{p}_2 \cdot \hat{q}_2}{n_2}}} \quad (3.98)$$

onde: \hat{p}_1 = valor estimado de p_1 .
 \hat{p}_2 = valor estimado de p_2 .
 $\hat{q}_1 = 1 - \hat{p}_1$.
 $\hat{q}_2 = 1 - \hat{p}_2$.
 n_1 = tamanho da amostra da população 1.
 n_2 = tamanho da amostra da população 2.

A hipótese nula poderá ser testada ao se comparar o valor de Z calculado pela expressão acima com um valor de Z crítico, que é determinado através de um nível de significância (máximo de probabilidade de cometer o erro tipo I, isto é, de rejeitar a hipótese H_0 quando esta é verdadeira). Para o caso deste trabalho, utilizou-se o nível de 5% de significância o que resultou em um Z igual a $\pm 1,96$, para o caso das hipóteses descritas acima.

Quando o Z calculado pela estatística estiver dentro do intervalo determinado pelo Z crítico, aceitamos a hipótese nula, que não há diferença significativa entre as proporções, caso contrário conclui-se ao nível de significância de 5% que existe diferença entre as proporções.

Utilizou-se este teste para comparar as proporções apresentadas em tabelas, no capítulo 5, que se referem aos resultados dos experimentos.

4. METODOLOGIA:

Descreve-se neste capítulo todo o desenvolvimento da parte experimental e o algoritmo elaborado no *software* MATLAB que compara os métodos selecionados para discriminação de cores reais em uma imagem.

O Método de Máxima Verossimilhança tem sido visto até aqui como um dos métodos estatísticos mais conhecidos e tido como muito eficaz para o reconhecimento de padrões em imagens de sensoriamento remoto (falsa-cor). Este método depende muito do vetor de médias e da matriz de covariância de cada classe espectral, modelada por uma distribuição normal multivariada (RICHARDS; LILLESAND & KIEFER; SONKA, HLAVAC & BOYLE; FOODY, McCULLOCH & YATES; HUANG; MAUSAL E PEDDLE).

Quando a classificação não depende da matriz de covariância das amostras utiliza-se o método de Distância Mínima, que é muito atrativo quando se deseja rapidez de processamento (RICHARDS, LILLESAND & KIEFER).

No entanto, o Método de Fisher assume que as classes a serem discriminadas não necessariamente precisam vir de uma população normal multivariada, ou seja, supõe-se que os dados vêm de uma mesma população qualquer, e que as matrizes de covariância de todas as classes sejam iguais (JOHNSON & WICHERN, 1985). Este método não aparece proposto em imagens de sensoriamento remoto (falsa-cor), pois sabe-se que não se pode pressupor que as classes venham de uma mesma população, mas que no caso de cores reais não se tem nenhuma restrição em sua utilização. Com isto, verifica-se com mais frequência a utilização do Método de Máxima Verossimilhança e de Distância Mínima.

Sendo a cor a variável a ser analisada neste trabalho, partiu-se de suas características para a escolha dos métodos estatísticos. O método de Máxima Verossimilhança foi escolhido devido a sua boa performance em estudos de reconhecimento de padrões de falsa-cor e espera-se que cores sejam caracterizadas como normalmente distribuídas. Outra característica da cor é que suas componentes possuem amplitudes pequenas, isto é, elas não podem variar muito em torno da média, implicando na mudança da sua tonalidade; tendo em vista este aspecto, optou-se pelo método de Fisher que considera as variâncias iguais e pelo método de Distância Mínima onde a variância é desconsiderada da função de discriminação.

Na tentativa de estudar mais a fundo a variável cor perante os métodos, escolheu-se analisar o comportamento das amostras que geram as funções discriminantes. Desta feita, controlou-se as amostras em algumas

características como média e variância, para verificar a performance dos métodos de uma maneira mais profunda.

No decorrer do trabalho, observou-se que os métodos de Máxima Verossimilhança e de Fisher consideravam que as componentes da cor seriam dependentes entre si. Foi então que, em busca de uma resposta quanto a este questionamento verificou-se que as componentes não possuíam esta dependência, pois de acordo com BANON (1989) estas se apresentam em um cubo unitário, no espaço referenciado pelo sistema de coordenadas ortogonais, o que leva a concluir que elas são independentes e portanto possuem covariância igual a zero. Assim, considerou-se esta hipótese e verificou-se a performance dos métodos alterando as matrizes de covariância, sendo que o método de Distância Mínima não foi afetado pela restrição, dado que na sua concepção utiliza somente os vetores de médias amostrais.

Outra questão seria a de verificar a diferença dos métodos quando se aumenta o número de cores a serem discriminadas, por exemplo, ao se comparar duas cores simultaneamente os métodos possuem um determinado desempenho que poderia piorar ao se acrescentar mais uma cor na comparação. Esta hipótese foi trabalhada não somente com três, mas com cinco, seis, dez e quinze cores discriminadas ao mesmo tempo.

O tempo de processamento foi analisado apenas a título de comparação entre os métodos, somente para se verificar qual apresenta maior rapidez, pois não houve preocupação, neste trabalho, em otimizar o algoritmo, dado que este foi concebido com única finalidade de comparar os métodos propostos.

Para comparar os métodos necessitou-se gerar imagens digitais coloridas controladas, que foi executado no *software* (MATLAB) com a caixa de ferramenta de Processamento de Imagem. Precisou-se também recorrer a caixa de ferramenta de Estatística deste mesmo *software*, para a comparação dos métodos.

Parte de uma foto digitalizada foi amostrada e classificada pelos métodos, para que se pudesse observar o desempenho dos mesmos frente a uma imagem cujas variáveis apresentassem caracter aleatório.

Todo processo de classificação e comparação dos métodos pode ser visto no esquema da figura 4.1 que se segue.

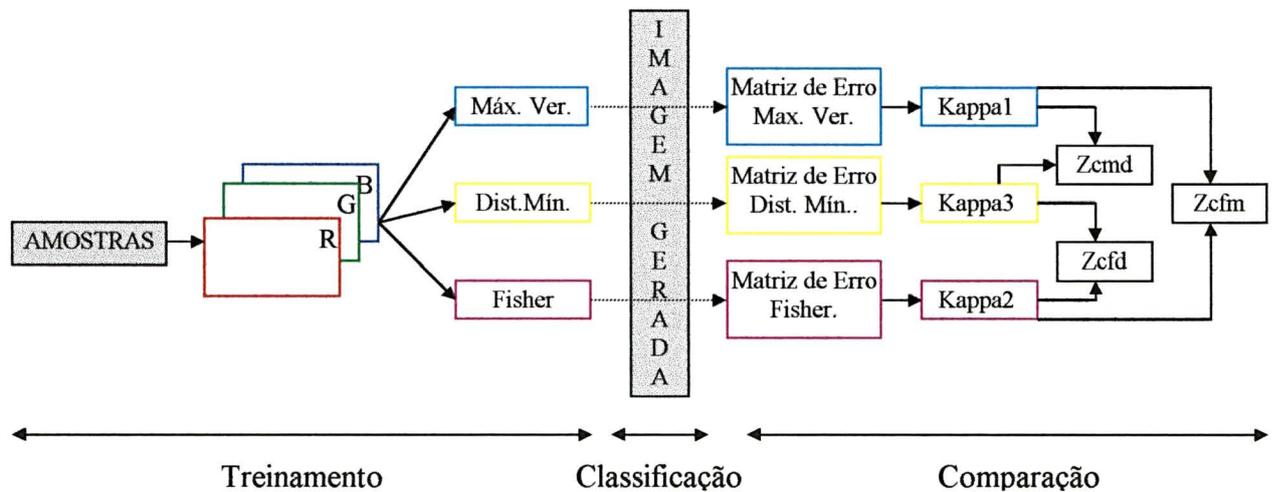
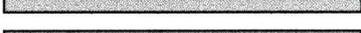
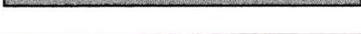


Figura 4.1- Etapas de Comparação dos Métodos.

4.1. CONSIDERAÇÕES GERAIS:

Com o intuito de testar as hipóteses apresentadas neste trabalho, a princípio reuniu-se um conjunto de 15 cores, apresentadas na tabela 4.1, donde selecionou-se 6 variações de verde, para simular cores que possam estar bem próximas umas das outras, e outras 10 cores chamadas de diversas que são encontradas comumente na natureza.

TABELA 4.1. Cores Definidas para Classificação.

	COR	R	G	B
1-		0	255	64
2-		0	255	128
3-		0	128	0
4-		0	128	64
5-		0	128	128
6-		0	255	0
7-		255	0	0
8-		0	0	255
9-		0	255	255
10-		255	255	0
11-		255	0	255
12-		0	0	0
13-		255	255	255
14-		192	192	192
15-		128	128	128

Estas cores foram classificadas duas a duas, três a três, e cinco a cinco para que se pudesse verificar o comportamento perante alguns aspectos apresentados neste capítulo. Também, utilizou-se as seis cores em tons de verde (1 a 6), as dez cores (6 a 15) e as quinze cores (1 a 15) classificadas conjuntamente. Para que estas comparações pudessem simular resultados concretos gerou-se imagens com 2, 3, 5, 6, 10 e 15 cores. Para gerar estas imagens criou-se matrizes onde cada elemento representa um pixel da imagem. O valor de cada elemento da matriz criada está relacionado a uma linha de outra matriz denominada de "mapa" que contém as componentes da cor (RGB). Isto quer dizer que a primeira matriz define a posição do pixel e o número constante nesta posição indica a linha correlata na segunda matriz (mapa), determinando a cor deste pixel (THOMPSON & SHURE, 1995).

Ao se comparar os métodos, analisou-se alguns aspectos que pudessem auxiliar no estudo do reconhecimento de imagens coloridas.

Levou-se em conta a natureza dos dados amostrais, como por exemplo a sua variação em torno da média, a média propriamente dita e o relacionamento entre as componentes (RGB) que compõem a cor.

Assim, a escolha dos métodos propostos deve-se a alguns fatores da natureza dos dados. O método de Máxima Verossimilhança considera que as variáveis se distribuem como normais multivariadas. Como se supõe que cada cor possui distribuição normal, optou-se por este método.

Ao se analisar a variação das componentes da cor, observou-se que para uma determinada cor, estas possuem amplitudes pequenas e iguais entre si, sugerindo uma variância constante entre as cores a serem discriminadas, que pode levar a pensar em dois métodos, o de Fisher e o de Distância Mínima. A concepção do método de Fisher é de que as cores venham de uma mesma população, sendo assim de mesma média e variância. O método de Distância Mínima, que é um caso particular do método de Máxima Verossimilhança considera que a variância entre as cores não afeta a discriminação sendo então eliminada da função e portanto um candidato a ter uma boa classificação.

Desta maneira, para que se pudesse observar o comportamento da variável aleatória tridimensional "cor" frente a sua discriminação, foi necessário um estudo experimental mais detalhado, com controle de algumas aspectos como média, variância e correlação entre as componentes da cor.

Assim, os três métodos aqui analisados foram comparados segundo estes aspectos acima mencionados para que se pudesse também verificar o comportamento da variável e com isto dar subsídios para a escolha de uma amostra de treinamento das classes (cores) quando for o caso.

Selecionou-se, então, vários tipos de amostras, chamadas de “mapas” no algoritmo, para que estes aspectos pudessem ser analisados.

Para a análise da interferência da média das amostras na classificação, verificou-se os seguintes casos:

1. Quando as médias são próximas dos valores a serem discriminados. Por exemplo, ao se discriminar a cor verde pura ($R=0$, $G=255$ e $B=0$), as amostras selecionadas para o treinamento são bem próximas dos valores das componentes, onde as variações de vermelho e azul são próximas de zero e a verde próxima de 255. Este tipo de média implica em uma variância pequena. Para diminuir a interferência, a variação das amostras entre as componentes foi feita de maneira homogênea, isto é, quando se variou a componente vermelha em uma unidade também se variou a verde e a azul em uma mesma unidade. Assim, chamou-se esta amostra de média próxima, variância pequena e homogênea. Para melhor entender, pode-se, por exemplo, ter uma amostra cujas componentes tenham $R = 1$, $G = 254$ e $B = 1$ e outra com $R = 2$, $G = 253$ e $B = 2$; sendo que a diferença entre elas é muito pequena resultando em uma média próxima aos valores das amostras.
2. Quando as médias das componentes das amostras são afastadas das componentes da cor a ser discriminada, com variância pequena e homogênea. Sabendo-se que este afastamento não foi muito acentuado, pois caso contrário se perderia a tonalidade da cor. Para exemplificar, pode-se ter uma amostra de cor com $R = 1$, $G = 254$ e $B = 1$; e outra com $R=11$; $G = 244$ e $B = 11$; então tem-se uma diferença entre elas de 10 unidades, resultando em uma média relativamente afastada das amostras.
3. Quando as médias das componentes das amostras tem afastamento variado resultando em uma variância heterogênea. Desta forma, pode-se ter, por exemplo, uma amostra com $R = 1$, $G = 250$ e $B = 11$ e outra com $R = 10$, $G = 254$ e $B = 5$; de onde se observa que as variações entre as componentes não foram as mesmas e que para outra amostra estas variaram de uma maneira diferente.
4. Quando as médias entre duas cores a serem discriminadas possuem tipos de médias diferentes, isto quer dizer que uma cor possui média do tipo próxima e outra do tipo afastada, ambas com variância pequena e homogênea.

Para os casos 1, 2 e 3 descritos acima, testou-se a existência da diferença entre 2, 6 e 15 cores simultaneamente comparadas. Para o caso de comparação entre 2 cores utilizou-se todas as cores disponíveis, no de 6 cores

verificou-se somente as tonalidades de verde (cores numeradas de 1 a 6), e no de 15 cores as cores denominadas de diversas que implica em médias bem diferentes. E para o caso 4, comparou-se somente as cores 2 a 2 para verificar o comportamento de duas cores com tipos de médias diferentes.

Outro aspecto de estudo neste trabalho foi o da influência da variância das amostras no desempenho dos métodos de classificação. Assim, selecionou-se quatro tipos de amostras da maneira como já descritas no caso de médias porém que tivessem as variâncias que seguem:

1. Com variâncias pequenas e homogêneas.
2. Com variâncias relativamente grandes e homogêneas.
3. Com variâncias heterogêneas.
4. Com variâncias diferentes entre duas cores a serem discriminadas.

Comparou-se a estes tipos de amostras nos casos de 2, 6 e 15 classificações simultâneas de cores, igualmente feito para o aspecto de médias.

É desejável, algumas vezes, que se selecione uma determinada cor, ou tons dela, em uma imagem que possua várias cores. Pergunta-se como se comportariam os métodos tendo-se em vista este aspecto?

Por esta razão, criou-se uma imagem contendo todas as 15 cores, de tal maneira que as tonalidades de verde (cores de 1 a 6) formassem as letras M, D e F ; como mostra a figura 4.2. Programou-se para que os métodos distinguíssem os tons de verde dos outros, e que quando encontrasse um tom verde colorisse de verde puro, referente a cor número 6 e para as demais, colorisse de branco referente a cor número 13, para que visualmente se desse a demonstração desta comparação. Com isto, pode-se verificar de uma maneira geral a comparação entre os métodos, onde, para este caso específico utilizou-se uma amostra com médias heterogêneas, como apresentadas nas figuras 5.1 e 5.3.

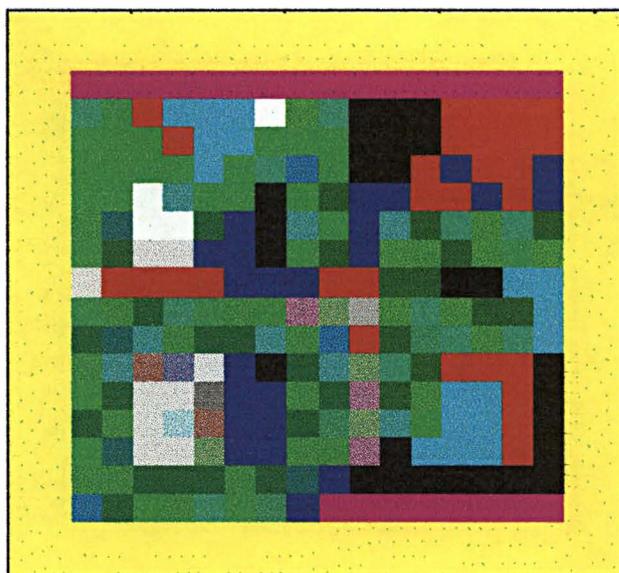


Figura 4.2 - Imagem Gerada para Identificação de Letras Verdes.

Na tentativa de estudar mais a fundo o comportamento dos métodos, observou-se que o método de Máxima Verossimilhança e o de Fisher consideravam as componentes da cor como dependentes entre si. Sabe-se, no entanto que estas componentes são independentes, pois segundo BANON (1989), a combinação das três componentes pode ser representada num espaço referenciado por um sistema de coordenadas ortogonais e que de acordo com SONKA, HLAVAC & BOYLE (1995), o sistema de processamento digital de imagem processa as componentes da cor independentemente considerando uma forte independência individual entre elas. Portanto, para que estes métodos considerassem a independência entre as componentes tiveram que sofrer uma modificação nas matrizes de covariância das cores, fazendo com que estas matrizes tivessem valores somente na diagonal principal.

Isso implica que as análises feitas neste trabalho fossem mostradas em dois grandes grupos: com dependência entre as componentes (métodos executados de modo original) e com independência entre as componentes (métodos modificados). No caso do método de Distância Mínima este não foi afetado pela restrição, pois este não utiliza as matrizes de covariância na discriminação.

Para que se pudesse obter resultados mais próximos da realidade, captou-se uma imagem digitalizada e selecionou-se uma parte desta, que continha três tons de cores predominantes (azul, vermelho e preto), para ser classificada pelos métodos. Cada pixel desta imagem foi visualmente classificado e uma amostra das cores foi selecionada para treinamento dos

métodos. Os resultados deste experimento estão apresentados no experimento 6 do capítulo 5 e a imagem completa da foto com a área selecionada aparece a seguir na figura 4.3.

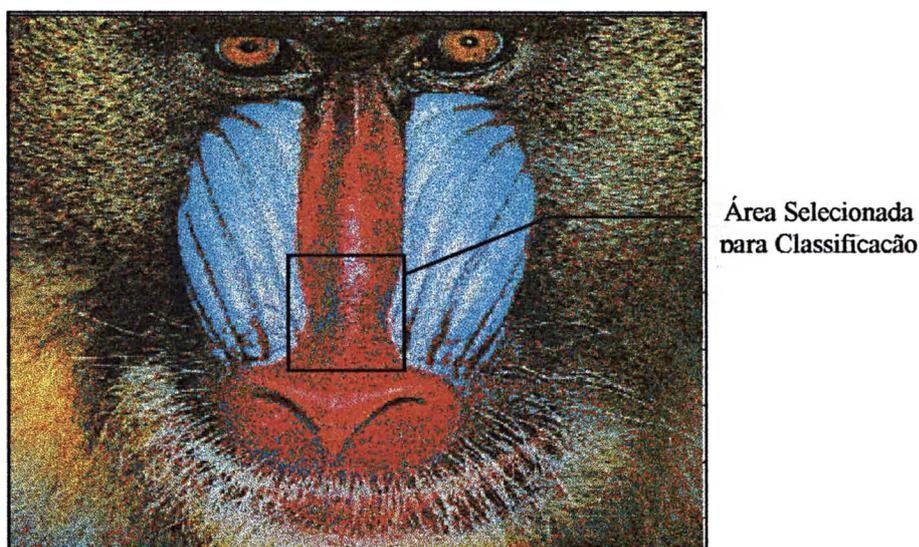


Figura 4.3 - Foto Digitalizada com a Área Selecionada para Classificação.

Um dos fatores que também podem afetar na escolha de um dos métodos para separação de cores é o tempo de processamento, principalmente quando esta precisa ser feita em tempo real (*on line*). Assim, introduziu-se no algoritmo um contador de tempo para cada método com intuito de fazer uma comparação entre eles. Neste caso específico, teremos o tempo médio de processamento de cada método a partir de discriminações entre 2, 3, 5, 6, 10 e 15 cores, onde pode-se examinar o fator número de cores separadas pelos métodos em relação ao tempo de processamento.

Toda parte experimental, foi feita, em um computador *Pentium-s* com 16M de memória RAM e velocidade de processamento de 133 Mhz.

No presente trabalho utilizou-se o *software* MATLAB com as caixas de ferramentas de Processamento de Imagem e de Estatística, com a ajuda das quais foi construído o algoritmo que possibilitou a comparação dos métodos, sendo que a primeira caixa, Processamento de Imagem, foi necessária para que se pudesse criar a imagem, retirar as matrizes que a compõe e visualizar tanto a imagem gerada, quanto as imagens de saída resultantes das classificações feitas pelos três métodos: Máxima Verossimilhança, Distância Mínima e Fisher. A outra, de Estatística, foi utilizada na fase de aplicação dos discriminantes e comparação dos métodos.

Na comparação dos métodos calculou-se as “Matrizes de Erros” e as estatísticas *kappa* que gera uma estatística Z calculada por uma expressão

enunciada no capítulo 4 e ao ser comparada com um Z crítico com nível de significância de 5% resulta na aceitação ou rejeição da hipótese de que existe ou não diferença entre os métodos.

Para que a análise dos resultados tivessem consistência, utilizou-se o teste de proporções, explicado no capítulo 4. Este teste compara um valor de Z calculado pela estatística com outro Z crítico com 5% de significância, assim se Z calculado estiver dentro do intervalo pelos valores de Z ($\pm 1,96$) aceita-se a hipótese nula que diz ser as proporções iguais, caso contrário há forte indício das proporções serem diferentes.

4.2. ALGORITMO:

Neste trabalho utilizou-se o *software* MATLAB, por este apresentar as ferramentas necessárias a execução do algoritmo de comparação entre os métodos (MATLAB User's Guide).

Este algoritmo foi feito com uma rotina de controle e várias sub-rotinas de execução, apresentadas no anexo I.

A rotina de controle foi denominada de “algodis” e mostra uma lista de opções (*menu*), como se observa na figura 4.4 a seguir.

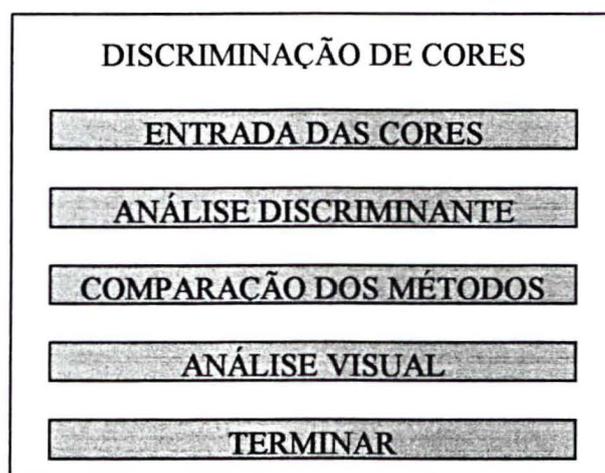


Figura 4.4: Opções do Algoritmo de Controle.

Assim, ao se selecionar uma das opções, obter-se-á uma sub-rotina que desempenhará a tarefa proposta.

A opção “Entrada de Cores”, abrirá outro *menu* onde a primeira opção se refere ao tipo de amostra em função da variação da média e da variância. e a segunda opção, designa o número de cores que se pretende discriminar concomitantemente. As figuras 4.5, 4.6 e 4.7 mostram as opções propostas

respectivamente para o entrada de cores, tipo de amostra e o número de cores.

ENTRADA DE CORES
TIPO DE AMOSTRA
No. DE CORES

Figura 4.5: Opções da Entrada de Cores.

TIPO DE AMOSTRA
Média Prox. e Var. Peq.Hom.
Média Afast. e Var. Peq. Hom.
Variância Grande
Médias Diferentes
Variâncias Diferentes

Figura 4.6: Opções do Tipo de Amostra.

No. DE CORES DISCRIMINADAS
2 CORES
3 CORES
5 CORES
6 CORES
10 CORES
15 CORES

Figura 4.7: Opções do No. de Cores

A opção Análise Discriminante, também abre um *menu* onde aparece em primeiro lugar uma sub-rotina que calcula as estatísticas comuns entre os métodos, que foi chamada de "Média e Variância", a seguir apresenta os três modelos de discriminação designadas por "Máxima Verossimilhança", "Distância Mínima" e "Fisher", como mostrado na figura 4.8.

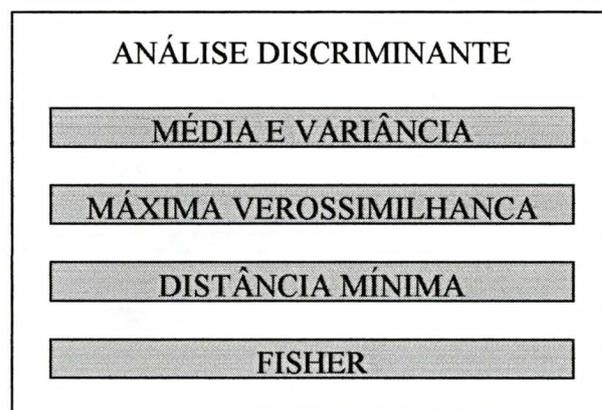


Figura 4.8: Opções da Análise Discriminante

A comparação dos métodos serão executadas quando for escolhida a opção "Comparação dos Métodos", que automaticamente calcula a Matriz de Erro e a estatística Kappa para cada método e ainda a estatística Z que compara os métodos e faz aparecer em tela o resultado, isto é, confirma se há ou não diferença entre os métodos.

Quando se deseja fazer uma verificação visual da análise escolhe-se a opção "Análise Visual", onde, através de alguns comandos com "imshow" e "subplot", aparecem lado a lado a figura da imagem original e as outras geradas pelos métodos discriminantes de Máxima Verossimilhança, Distância Mínima e Fisher, como mostra a figura 4.9 a seguir, nas cores 1 e 6 (respectivamente verde escuro e verde puro)(THOMPSON & SHURE, 1995).

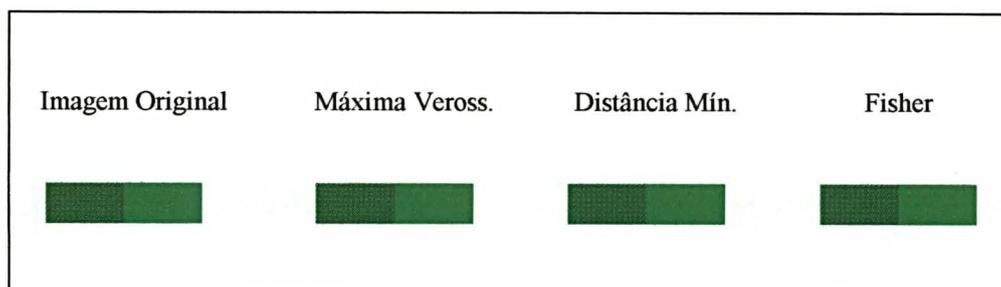


Figura 4.9: Análise Visual do Desempenho dos Métodos.

Em qualquer tempo que se deseja sair do programa, escolhe-se a opção "Terminar", onde aparece a pergunta: "Deseja sair do programa?" com as seguintes opções de resposta: "Sim, salvando variáveis", "Sim, sem salvar variáveis", "Não, salvando variáveis" e "Não, sem salvar variáveis". Com isto, se for selecionado salvar variáveis esta será feita em um arquivo chamado "discr.dat". Pode-se ver na figura 4.10 a seguir estas opções da maneira que aparecem na caixa de ferramenta.

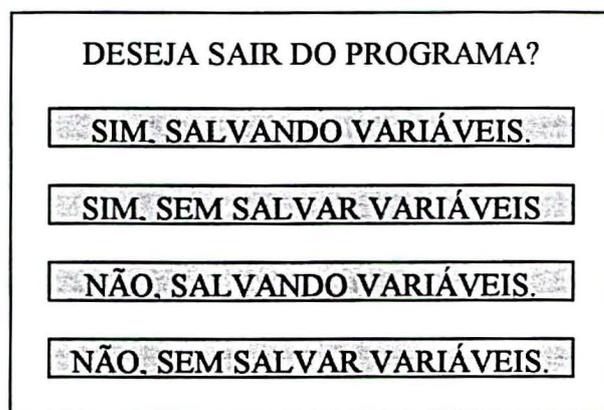


Figura 4.10: Opções da Saída do Programa.

O esquema mostrado na figura 4.11, resume toda a seqüência do algoritmo, incluído da denominação dada às subrotinas no programa.

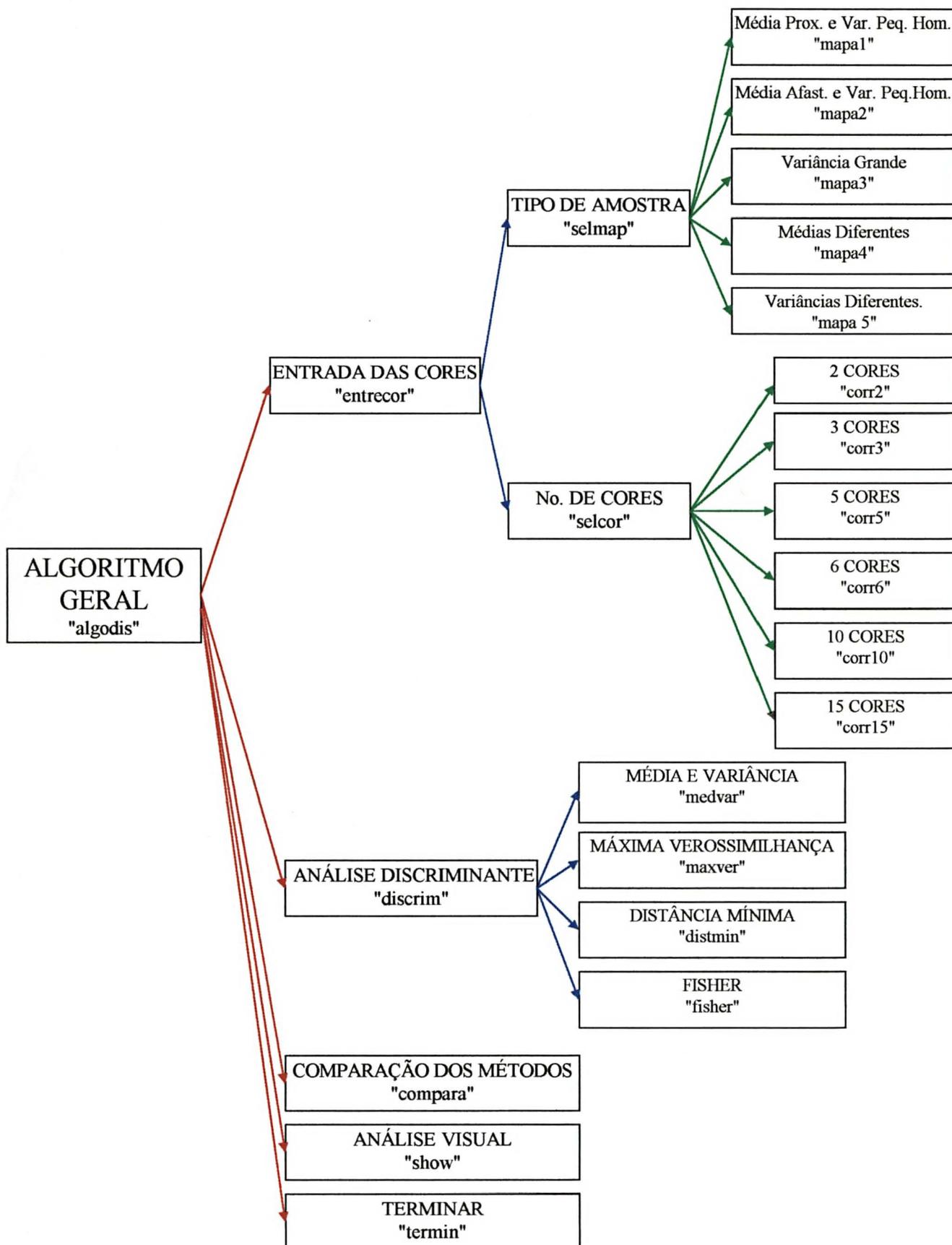


Figura 4.11: Esquema do Programa.

O capítulo a seguir apresenta os resultados da aplicação do algoritmo.

5. ANÁLISE DOS RESULTADOS.

Neste capítulo apresenta-se os resultados obtidos dos experimentos feitos a partir de alguns aspectos que poderiam afetar a performance dos métodos propostos para reconhecimento de padrões de cores de uma imagem.

Os aspectos mencionados no item 4.1 deste trabalho foram abordados aqui em seis experimentos, onde o experimento 1 se refere à média das amostras, o experimento 2 à sua variância, o experimento 3 ao número de cores a serem concomitantemente discriminadas, o experimento 4 ao tempo de processamento dos métodos, o experimento 5 separa uma tonalidade de cor de uma imagem que contém 15 cores diferentes e o experimento 6 classifica duas tonalidades de cores analisadas em uma foto digitalizada comparando os métodos propostos em um caso real.

No entanto, estes seis experimentos foram duplamente executados, sendo que no primeiro caso considerou-se a existência de correlação entre as componentes espectrais (RGB) e no segundo caso supôs-se ser as componentes independentes uma das outras.

Segue nos itens deste capítulo estas duas abordagens do problema, contendo cada uma os sub-itens relacionados aos experimentos expostos acima.

5.1. 1º. CASO.

Os métodos, neste item, apresentam-se de modo original, sem alteração na matriz de covariância das amostras, visto que se considerou neste caso a existência de dependência entre as componentes das cores (RGB).

5.1.1. Experimento 1.

O objetivo principal deste experimento foi submeter os métodos a vários tipos de médias das amostras de treinamento, verificando de que maneira estas afetam o desempenho do discriminante.

Selecionou-se quatro tipos de amostras, descritas no item 4.1, que se diferenciam em alguns pontos apresentados a seguir.

Duas destas amostras se referem ao afastamento em torno do valor da cor original a ser discriminada, criando assim uma amostra com média próxima a este valor e outra com média afastada a ele, ambas com variância pequena e homogênea.

O segundo ponto a ser analisado se refere a homogeneidade das amostras. Entende-se como amostra homogênea aquela em que todas as componentes da cor variem igualmente, isto é, quando a componente vermelha de uma cor variar em 5 unidades, verde e azul também variam em 5 unidades. Isto representa dizer que a variância é constante na componentes e por conseguinte no caso heterogêneo intui-se que exista uma diferenciação nas variâncias, de tal maneira que a média também sofra variação e de alguma maneira isto afetaria os resultados gerados pela aplicação dos métodos.

Outro fator a considerar foi observar se a performance dos métodos foi afetada ao se classificar duas cores, sendo que estas possuem tipos de médias diferenciadas, ou seja, uma cor possui média próxima e outra tenha média afastada da cor de controle.

Os resultados destes experimentos podem ser vistos na tabela 5.1, onde se tem o número e percentual de erros de classificação. Nas linhas observa-se o tipo de amostra referentes às médias, definidas no experimento, e nas colunas os três métodos propostos cuja classificação foi feita com duas cores em dois grupos: tons de verde que correspondem às cores numeradas de 1 a 6 e totalizam 15 combinações distintas e cores diversas de 6 a 15 com 45 combinações, com 6 cores onde se utilizou somente as cores de tons verde e 15 cores que incluem todas as cores estudadas.

Ao analisar a tabela, observou-se que o método de Distância Mínima não apresenta nenhum erro, independente do tipo da média das amostras utilizadas, apesar deste método se basear unicamente na média amostral.

O método de Máxima Verossimilhança apresenta uma porcentagem de erros muito elevada, chegando em alguns casos a totalizar 100% de classificação errada, e melhorando o seu desempenho somente nas amostras com média heterogênea.

No caso do método de Fisher, percebe-se que este mostrou uma boa performance e não apresentou diferença significativa comparado ao método de Distância Mínima.

Realizou-se o teste estatístico de proporções, referenciado no capítulo 4, para verificar se a média das amostras estaria influenciando nos métodos. Verificou-se uma diferença significativa a um nível de significância de 5% ($Z = 3,435$ para duas cores de tons de verde, $Z = 20,97$ para duas cores tons diversos, $Z = 3,17$ para as 15 cores comparado com Z crítico de $\pm 1,96$) quanto a homogeneidade das amostras para o método de Máxima Verossimilhança. Não apresentando, portando, nenhuma diferença nos outros aspectos da média aqui estudados.

Tabela 5.1 Percentual de Erros de Classificação por Tipo de Média Amostral Considerando Dependência entre as Componentes

TIPOS DE AMOSTRAS		MÉTODOS														
		MÁXIMA VEROSS.					DISTÂNCIA MÍNIMA					FISHER				
		2 Cores		6 Cores		15 Cores	2 Cores		6 Cores		15 Cores	2 Cores		6 Cores		15 Cores
		Ver.	Div.	Cores	Cores	Ver.	Div.	Cores	Cores	Ver.	Div.	Cores	Cores	Ver.	Div.	Cores
1. Média Próxima Homogênea	Erro/Total	14/15	45/45	5/6	14/15	0/15	0/45	0/6	0/15	0/15	2/45	3/6	0/15	0/15	0/15	0/15
	%	93	100	83	93	0	0	0	0	0	4	50	0	0	0	0
2. Média Afastada Homogênea	Erro/Total	12/15	42/45	5/6	14/15	0/15	0/45	0/6	0/15	0/15	0/45	3/6	0/15	0/15	0/15	0/15
	%	80	93	83	93	0	0	0	0	0	0	50	0	0	0	0
3. Média Heterogênea	Erro/Total	4/15	1/45	3/6	7/15	0/15	0/45	0/6	0/15	3/15	0/45	0/6	0/15	3/15	0/45	0/15
	%	27	2	50	47	0	0	0	0	20	0	0	0	20	0	0
4. Médias Diferentes	Erro/Total	30/30	90/90	-	-	0/30	0/90	-	-	1/30	0/90	-	-	1/30	0/90	-
	%	100	100	-	-	0	0	-	-	3	0	-	-	3	0	-

5.1.2 Experimento 2.

Na tentativa de verificar se a variância estaria influenciando na performance dos métodos, selecionou-se quatro amostras que representassem todos os tipos de variações que possam ocorrer na prática.

Essas amostras mencionadas no item 4.1 possuem três características trabalhadas neste item. A primeira se refere a variações em torno da média ser pequena ou relativamente grande, sendo ambas de concepção homogênea, isto é, constante entre as componentes espectrais. A segunda, identifica se a homogeneidade das variações entre as componentes afeta a classificação. A última característica tenta identificar se ao classificar duas cores que apresentem em suas amostras variâncias diferentes, ou seja, uma com variância pequena e outra grande, respondam igualmente à classificação.

Na tabela 5.2 mostrou-se os erros de classificação dos três métodos propostos, que aparecem nas classificações de tonalidade de verde (que totalizam seis) tomadas duas a duas, as cores chamadas de diversas (que totalizam dez) tomadas duas a duas, as seis tonalidades de verde conjuntamente observadas e as quinze cores propostas ao mesmo tempo, sendo que para cada um dos casos introduziu-se amostras que contivessem variâncias dos tipos: homogênea pequena, homogênea grande, heterogênea. No caso específico de classificação de cores duas a duas, realizou-se uma experimentação com intuito de verificar a resposta dos classificadores frente a duas cores com variâncias diferentes.

O método de Distância Mínima não apresentou diferença de classificação entre os vários tipos de variâncias, o que era de se esperar visto que este método não utiliza a matriz de covariância em sua função discriminante. Observa-se também que novamente não ocorreram erros de classificação em nenhum dos casos, o que indica que é um método poderoso para classificação de cores.

Como já havia ocorrido no experimento 1, o método de Máxima Verossimilhança apresentou muitos erros de classificação tendo melhorado um pouco quando submeteu-se o classificador a uma amostra com variância heterogênea, isto é, variações diferentes entre as componentes espectrais de cada cor.

No caso do método de Fisher, notou-se uma performance razoável, mas não ótima como o método de Distância Mínima.

Tabela 5.2 Percentual de Erros de Classificação por Tipo de Variância Amostral Considerando Dependência entre as Componentes

TIPOS DE AMOSTRAS		MÉTODOS														
		MÁXIMA VEROSS.					DISTÂNCIA MÍNIMA					FISHER				
		2 Cores		6		15	2 Cores		6		15	2 Cores		6		15
Ver.	Div.	Cores	Cores	Cores	Ver.	Div.	Cores	Cores	Cores	Ver.	Div.	Cores	Cores	Cores		
1. Variância Pequena Homogênea	Erro/Total	14/15	45/45	5/6	14/15	14/15	0/45	0/6	0/6	0/15	0/15	2/45	3/6	0/15		
	%	93	100	83	93	93	0	0	0	0	0	4	50	0		
2. Variância Grande Homogênea	Erro/Total	9/15	42/45	5/6	14/15	14/15	0/45	0/6	0/6	0/15	0/15	0/45	3/6	0/15		
	%	60	93	83	93	93	0	0	0	0	0	0	50	0		
3. Variância Heterogênea	Erro/Total	4/15	1/45	3/6	7/15	7/15	0/45	0/6	0/6	0/15	0/15	0/45	0/6	0/15		
	%	27	2	50	47	47	0	0	0	0	0	20	0	0		
4. Variâncias Diferentes	Erro/Total	30/30	90/90	-	-	-	0/30	0/90	-	-	0/30	0/90	-	-		
	%	100	100	-	-	-	0	0	-	-	0	0	-	-		

Ao submeter-se os dados ao teste de proporções, descrito no item 3.2.2, verificou-se que variâncias pequenas e grandes não apresentaram diferenças significativas no método de Fisher, apesar de se observar uma deficiência de classificação muito grande quando se comparou as tonalidades de verde (50% de classificações erradas). Confrontou-se, ainda neste método, as porcentagens das amostras com variância homogênea e heterogênea e ao nível de significância de 5% que corresponde a intervalo de Z crítico de $[-1,96; +1,96]$, registrou-se um Z calculado pela estatística do teste de 1,9365, donde se conclui que não há diferença significativa nestes tipos de amostras para este método.

No entanto, o método de Máxima Verossimilhança apresentou diferenças significativas no que se refere a diferença de variâncias tipo grande e pequeno, nas comparação de amostra de tons de verde tomados duas a duas ($Z=2,31$), e nas variâncias do tipo homogênea e heterogênea ao se classificar as cores diversas duas a duas ($Z=20,97$) e todas ao mesmo tempo ($Z=3,17$).

5.1.3. Experimento 3.

Uma das hipóteses da pesquisa é verificar se o desempenho dos métodos ficaria prejudicado ao aumentar o número de cores a serem discriminadas. Escolheu-se para tanto a amostra de média e variância heterogênea.

A tabela 5.3 apresenta este resultado, sendo que quando possível introduziu-se os dois conjuntos de cores propostos no capítulo 4, que seriam as cores de tonalidade verde, representando cores que poderiam estar bem próxima uma das outras e as cores que se denominou de diversas (vermelho, azul, amarelo, etc.).

Nota-se que o número de cores simultaneamente discriminadas não alterou em nada o método de Mínima Distância, visto que para qualquer quantidade não houve sequer um erro de classificação.

O método de Fisher apresentou 20% de erros ao se discriminar as tonalidades de verde duas a duas, mas quando se aumentou para uma comparação três a três este percentual de erro caiu para 5%, e de cinco em cinco em diante caiu para 0%; no entanto, para as cores diversas observou-se que quando se comparada as cores de três em três, e somente neste caso, obteve-se um percentual de erro na ordem de 7%, pois para os

outros casos o percentual permaneceu em 0%. Uma provável causa desta ocorrência pode ter sido ocasionada pela própria concepção do método.¹

O desempenho do método de Máxima Verossimilhança também apresentou considerável percentual de erros ao se comparar três cores ao mesmo tempo, tanto no caso de tonalidades de verde como no de cores diversas. E no restante, observa-se que para tonalidades de verde o método piora até a comparação de 5 cores, melhorando posteriormente; e nas diversas tonalidades de cores aumenta o percentual de erros de 2 cores para 3 cores e em seguida este valor cai para 5 cores e sobe de uma maneira não significativa para 15 cores.

Analisando-se os resultados de uma maneira geral, vê-se que quanto mais cores se discriminar num mesmo instante, há uma melhora no desempenho dos métodos de Máxima Verossimilhança e de Fisher, contrariando o esperado.

5.1.4. Experimento 4.

Um dos fatores de suma importância nos dias de hoje é o tempo de processamento dos métodos, pela necessidade crescente de reconhecer padrões de modo instantâneo (on line). Não foi preocupação neste trabalho desenvolver este tema, mas pode-se oferecer uma pequena contribuição que venha auxiliar trabalhos futuros.

Foi feita aqui apenas uma comparação entre o tempo de processamento, para definir o método mais rápido, não houve no entanto otimizações de algoritmos para que o tempo fosse melhorado.

Desta maneira, observou-se na tabela 5.4 a média dos tempos coletados de um contador de tempo do *software* MATLAB para os diversos números de cores processadas conjuntamente, salientando que a imagem de reconhecimento foi gerada com o mesmo número de pixel, para que o processo ficasse livre desta variável na comparação.

¹ O método de Fisher, segundo Johnson & Wichern (1985), possui uma restrição no número de funções discriminantes, pois o número de funções é igual ao número de autovalores diferentes de zero. Assim, teria-se o número de funções iguais ao mínimo entre o número de variáveis (p) e o número de classes (g) menos 1, ou seja, $\min(p, g-1)$. No caso estudado ter-se-á, p igual a 3 (RGB) e g igual a 3, resultando um total de duas funções. Este resultado, não deveria afetar a classificação, pois segundo os autores, não há perda de informações com a redução da dimensão.

Tabela 5.3 - Total e Percentual de Erros de Classificação por No. de Cores Discriminadas Simultaneamente Considerando Dependência entre as Componentes.

NÚMERO DE CORES DISCRIMINADAS	MÉTODO											
	MÁXIMA VEROSS.			DISTÂNCIA MÍNIMA			FISHER					
	Erro/Total	%		Erro/Total	%		Erro/Total	%				
2 Cores	Verd.	4/15	27	0/15	0		3/15	20				
	Diver.	1/15	2	0/15	0		0/15	0				
3 Cores	Verd.	14/20	70	0/20	0		1/20	5				
	Diver.	120/120	100	0/120	0		8/120	7				
5 Cores	Verd.	6/6	100	0/6	0		0/6	0				
	Diver.	56/252	22	0/252	0		0/252	0				
6 Cores	Verd.	2/6	33	0/6	0		0/6	0				
	Diver.	-		-	-		-	-				
10 Cores	Verd.	-		-	-		-	-				
	Diver.	0/10	0	0/10	0		0/10	0				
15 Cores	Verd.e Diver.	4/15	27	0/15	0		0/15	0				

Tabela 5.4 - Tempo Médio de Processamento por Método e No. de Cores Simultaneamente Discriminadas Considerando Dependência entre as Componentes.

NÚMERO DE CORES DISCRIMINADAS	MÉTODOS								Total	
	MÁXIMA VEROSS.		DISTÂNCIA MÍNIMA		FISHER		Total		Tempo (seg)	%
	Tempo (seg)	%	Tempo (seg)	%	Tempo (seg)	%	Tempo (seg)	%		
2 Cores	0,38	59	0,06	10	0,20	31	0,64	100		
3 Cores	0,83	63	0,12	9	0,36	27	1,31	100		
5 Cores	1,50	66	0,24	11	0,52	23	2,26	100		
6 Cores	1,58	63	0,29	12	0,59	25	2,49	100		
10 Cores	2,35	64	0,40	11	0,92	25	3,67	100		
15 Cores	3,67	65	0,55	10	1,32	25	5,61	100		

Tabela 5.6 : Tabela de Contingência - Método de Distância Mínima Considerando Dependência entre as Componentes.

		DADOS CLASSIFICADOS														
D	COR	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
A	1	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	2	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0
O	3	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0
S	4	0	0	0	35	0	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0
R	6	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0
E	7	0	0	0	0	0	0	29	0	0	0	0	0	0	0	0
F	8	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0
E	9	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0
R	10	0	0	0	0	0	0	0	0	0	144	0	0	0	0	0
Ê	11	0	0	0	0	0	0	0	0	0	0	24	0	0	0	0
N	12	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0
C	13	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0
I	14	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0
A	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12

Tabela 5.7: Tabela de Contingência - Método de Fisher Considerando Dependência entre as Componentes.

		DADOS CLASSIFICADOS														
D	COR	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
A	1	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	2	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0
O	3	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0
S	4	0	0	0	35	0	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0
R	6	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0
E	7	0	0	0	0	0	0	29	0	0	0	0	0	0	0	0
F	8	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0
E	9	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0
R	10	0	0	0	0	0	0	0	0	0	144	0	0	0	0	0
Ê	11	0	0	0	0	0	0	0	0	0	0	24	0	0	0	0
N	12	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0
C	13	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0
I	14	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0
A	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12

O Método de Máxima Verossimilhança não obteve bom desempenho, tanto que na análise estatística das tabelas o valor de *kappa* resultou em 0,7637, o que corresponderia a dizer que o método teve uma performance 76,37% de classificações corretas, contra 100% dos outros métodos (Distância Mínima e Fisher).

O teste resultou em um valor igual de Z (-2,1015) na comparação entre o método de Máxima Verossimilhança com os dois outros métodos,

observa-se que este valor se situa fora do intervalo $[-1,96; +1,96]$, o que implica em uma rejeição da hipótese de que os métodos seriam iguais.

Pode-se ver ainda que ao se analisar as imagens geradas pela aplicação dos métodos de modo visual (figura 5.1) observa-se claramente que as letras não aparecem em sua totalidade quando se classificou a imagem pelo método de Máxima Verossimilhança, porém no caso dos outros métodos isto não ocorreu.

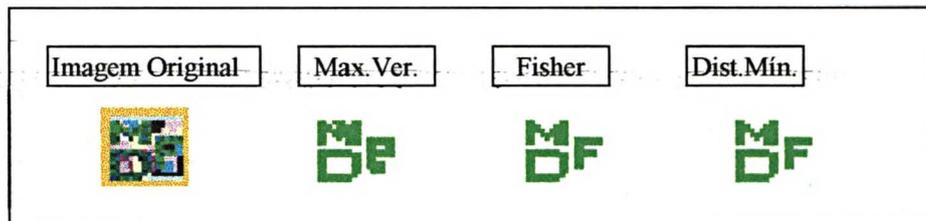


Figura 5.1 - Análise Visual da Comparação dos Métodos com Letras Considerando Componentes Dependentes.

5.1.6. Experimento 6.

Parte de uma foto digitalizada no *software* MATLAB foi submetida ao algoritmo, com intuito de analisar os métodos em um caso real. Assim, extraiu-se amostras das três cores (azul, vermelho e preto) predominantes na imagem digitalizada e processou-se os métodos, donde resultou a figura 5.2.

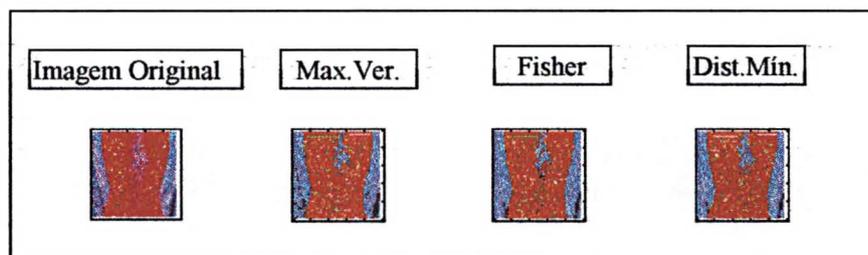


Figura 5.2 - Análise Visual da Comparação dos Métodos de Parte da Foto Digitalizada Considerando Componentes Dependentes.

Os resultados estatísticos aparecem nas tabelas de contingência 5.8, 5.9 e 5.10 a seguir que correspondem cada uma aos métodos de Máxima Verossimilhança, Distância Mínima e Fisher respectivamente.

Tabela 5.8: Tabela de Contingência - Método de Máxima Verossimilhança Considerando Dependência entre as Componentes para Imagem Digitalizada.

DADOS DE REFERÊNCIA	DADOS CLASSIFICADOS				
	COR	Vermelho	Preto	Azul	TOTAL
Vermelho		6.076	241	436	6.753
Preto		6	48	0	54
Azul		217	0	2.976	3.193
TOTAL		6.299	289	3.412	10.000

Tabela 5.9: Tabela de Contingência - Método de Distância Mínima Considerando Dependência entre as Componentes para Imagem Digitalizada.

DADOS DE REFERÊNCIA	DADOS CLASSIFICADOS				
	COR	Vermelho	Preto	Azul	TOTAL
Vermelho		6.070	70	613	6.753
Preto		2	52	0	54
Azul		129	0	3.064	3.193
TOTAL		6.201	122	3.677	10.000

Tabela 5.10: Tabela de Contingência - Método de Fisher Considerando Dependência entre as Componentes para Imagem Digitalizada.

DADOS DE REFERÊNCIA	DADOS CLASSIFICADOS				
	COR	Vermelho	Preto	Azul	TOTAL
Vermelho		5.993	147	613	6.753
Preto		2	52	0	54
Azul		217	16	2.960	3.193
TOTAL		6.212	215	3.573	10.000

Observa-se com surpresa que neste caso específico, onde tudo é aleatório os três métodos apresentam performance diferentes, confirmado pelo teste Z, descrito no item 3.2.1, ao nível de significância de 5% e pelo cálculo da estatística *kappa* de cada método, que apresentou os resultados de 0,8067, 0,8245 e 0,7866 para os métodos de Máxima Verossimilhança, Distância Mínima e Fisher respectivamente. Portanto, verifica-se que o método de Distância Mínima apresentou neste caso uma melhor performance, seguido do método de Máxima Verossimilhança e do método de Fisher.

5.2. 2º: CASO.

Ao se introduzir a independência entre as componentes da cor, ou seja, quando se alterou as matrizes de covariância de tal maneira que se considerou somente os valores obtidos na diagonal principal, tornando zero os outros valores; observou-se que todos os métodos apresentaram solução ótima, isto é, não houve erro nos experimentos realizados. As tabelas 5.11, 5.12 e 5.13 apresentam os resultados obtidos dos experimentos 1, 2 e 3.

Quanto ao tempo de processamento total os métodos com a restrição de correlacionamento mostraram-se cerca de 20% mais rápidos do que os métodos sem esta restrição, o que deve a maior facilidade de cálculo.

Verifica-se na tabela 5.14 que as proporções obtidas na tabela 5.4 para componentes correlacionadas, não sofreram alterações a não ser no método de Máxima Verossimilhança onde apresentou um tempo mais rápido quando comparou duas cores, porém ao processar mais cores conjuntamente voltou a obter o percentual observado no outros casos.

Desta maneira, a método de Máxima Verossimilhança utilizou cerca de 54% do tempo total de processamento utilizado para discriminação, o de Distância Mínima por volta de 13% e o de Fisher em torno de 33%.

Os resultados apresentados no experimento 5, descrito no capítulo 4, que se refere a classificação de uma cor retirada de uma imagem contendo várias tonalidades, implicam em uma classificação de 100% correta para todos os métodos e pode-se ver isto claramente na figura 5.3 cuja imagem foi gerada para análise visual.

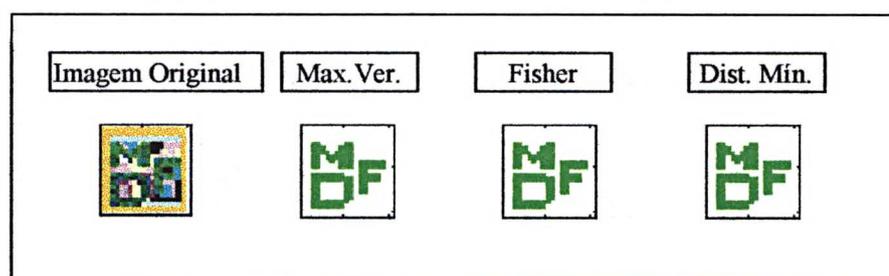


Figura 5.3 - Análise Visual da Comparação dos Métodos da Imagem com Letras Considerando Componentes Independentes.

Tabela 5.11 Percentual de Erros de Classificação por Tipo de Média Amostral Considerando Independência entre as Componentes

TIPOS DE AMOSTRAS		MÉTODOS														
		MÁXIMA VEROSS.				DISTÂNCIA MÍNIMA				FISHER						
		2 Cores		6		15		6		15		2 Cores		6		15
Ver.	Div.	Cores	Cores	Ver.	Div.	Cores	Cores	Ver.	Div.	Cores	Cores	Ver.	Div.	Cores	Cores	
1. Média Próxima Homogênea	Erro/Total	0/15	0/45	0/6	0/15	0/15	0/6	0/6	0/45	0/45	0/15	0/15	0/15	0/45	0/6	0/15
	%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2. Média Afastada Homogênea	Erro/Total	0/15	0/45	0/6	0/15	0/15	0/6	0/6	0/45	0/45	0/15	0/15	0/15	0/45	0/6	0/15
	%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3. Média Heterogênea	Erro/Total	0/15	0/45	0/6	0/15	0/15	0/6	0/6	0/45	0/45	0/15	0/15	0/15	0/45	0/6	0/15
	%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4. Médias Diferentes	Erro/Total	0/30	0/90	-	-	0/30	0/90	-	0/30	0/90	-	-	0/30	0/90	-	-
	%	0	0	-	-	0	0	-	0	0	-	-	0	0	-	-

Tabela 5.12 - Percentual de Erros de Classificação por Tipo de Variância Considerando Independência entre as Componentes.

TIPOS DE AMOSTRAS		MÉTODOS															
		MÁXIMA VEROSS.						DISTÂNCIA MÍNIMA						FISHER			
		2 Cores		6		15		2 Cores		6		15		2 Cores		6	15
		Ver.	Div.	Cores	Cores	Ver.	Div.	Cores	Cores	Ver.	Div.	Cores	Cores	Ver.	Div.	Cores	
1. Variância Pequena Homogênea	Erro/Total	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/15
	%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2. Variância Grande Homogênea	Erro/Total	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/15
	%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3. Variância Heterogênea	Erro/Total	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/6	0/15	0/45	0/6	0/15
	%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4. Variâncias Diferentes	Erro/Total	0/30	0/90	-	-	0/30	0/90	-	-	0/30	0/90	-	-	0/30	0/90	-	-
	%	0	0	-	-	0	0	-	-	0	0	-	-	0	0	-	-

Tabela 5.13 - Total e Percentual de Erros de Classificação por No. de Cores Discriminadas Simultaneamente Considerando Independência entre as Componentes.

NÚMERO DE CORES DISCRIMINADAS	MÉTODO											
	MÁXIMA VEROSS.			DISTÂNCIA MÍNIMA			FISHER					
	Erro/Total	%		Erro/Total	%		Erro/Total	%				
2 Cores	Verd.	0/15	0	0/15	0		0/15	0				
	Diver.	0/15	0	0/15	0		0/15	0				
3 Cores	Verd.	0/20	0	0/20	0		0/20	0				
	Diver.	0/120	0	0/120	0		0/120	0				
5 Cores	Verd.	0/6	0	0/6	0		0/6	0				
	Diver.	0/252	0	0/252	0		0/252	0				
6 Cores	Verd.	0/6	0	0/6	0		0/6	0				
	Diver.	-		-			-					
10 Cores	Verd.	-		-			-					
	Diver.	0/10	0	0/10	0		0/10	0				
15 Cores	Verd.e Diver.	0/15	0	0/15	0		0/15	0				

Tabela 5.14 - Tempo Médio de Processamento por Método e No. de Cores Simultaneamente Discriminadas Considerando Independência entre as Componentes.

NÚMERO DE CORES DISCRIMINADAS	MÉTODO								Total	
	MÁXIMA VEROSS.		DISTÂNCIA MÍNIMA		FISHER				Tempo (seg)	%
	Tempo (seg)	%	Tempo (seg)	%	Tempo (seg)	%	Tempo (seg)	%		
2 Cores	0,12	29	0,08	19	0,22	52	0,22	52	0,42	100
3 Cores	0,55	57	0,12	12	0,30	31	0,30	31	0,97	100
5 Cores	1,06	59	0,22	12	0,52	29	0,52	29	1,80	100
6 Cores	1,21	59	0,26	12	0,59	29	0,59	29	2,06	100
10Cores	1,81	57	0,47	15	0,89	28	0,89	28	3,17	100
15 Cores	2,81	59	0,53	12	1,38	29	1,38	29	4,72	100

Submetendo-se os métodos a um caso real (experimento 6), obteve-se os resultados apresentados nas tabelas de contingência que seguem:

Tabela 5.15: Tabela de Contingência - Método de Máxima Verossimilhança Considerando Independência das Componentes para Imagem Digitalizada.

DADOS DE REFERÊNCIA	DADOS CLASSIFICADOS				
	COR	Vermelho	Preto	Azul	TOTAL
Vermelho		5.076	1.547	130	6.753
Preto		2	52	0	54
Azul		0	359	2.834	3.193
TOTAL		5.078	1.958	2.964	10.000

Tabela 5.16: Tabela de Contingência - Método de Distância Mínima Considerando Independência das Componentes para Imagem Digitalizada.

DADOS DE REFERÊNCIA	DADOS CLASSIFICADOS				
	COR	Vermelho	Preto	Azul	TOTAL
Vermelho		6.070	70	613	6.753
Preto		2	52	0	54
Azul		129	0	3.064	3.193
TOTAL		6.201	122	3.677	10.000

Tabela 5.17: Tabela de Contingência - Método de Fisher Considerando Independência das Componentes para Imagem Digitalizada.

DADOS DE REFERÊNCIA	DADOS CLASSIFICADOS				
	COR	Vermelho	Preto	Azul	TOTAL
Vermelho		6.070	70	613	6.753
Preto		2	52	0	54
Azul		111	0	3.082	3.193
TOTAL		6.183	122	3.695	10.000

Observa-se os resultados da análise visual na figura 5.4 a seguir.

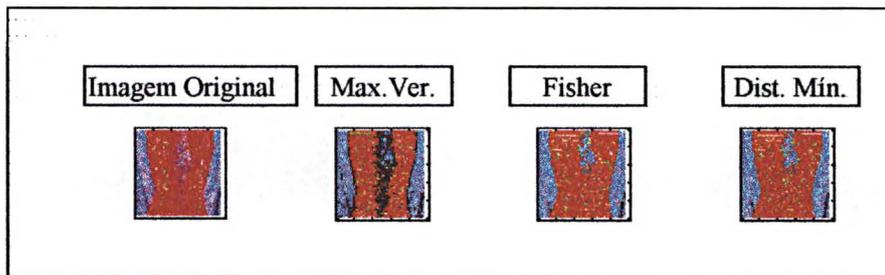


Figura 5.4 - Análise Visual da Comparação dos Métodos de Parte da Foto Digitalizada Considerando Componentes Independentes.

Analisando a estatística *kappa* de cada um dos métodos resultou em 0,6370 para o método de Máxima Verossimilhança, 0,8245 para o método de Distância Mínima e 0,8286 para o método de Fisher; e ainda fazendo o teste *Z*, observa-se que existe diferença significativa a nível de 5% do método de Máxima Verossimilhança com os outros métodos, verificando-se também que este piorou se comparado ao experimento semelhante que considerava a dependência entre as componentes da cor.

As conclusões são apresentadas no capítulo a seguir, bem com as recomendações.

6. CONCLUSÕES E RECOMENDAÇÕES.

O estudo de comparação dos métodos estatísticos analisados proporcionou, ao se tentar controlar o processo, subsídios para identificar aspectos das amostras. Estes aspectos podem ser importantes quando não se tem amostras e se deseja criá-las.

Percebeu-se que os métodos de Máxima Verossimilhança e Fisher são susceptíveis ao tipo de amostra selecionada, porém o método de Distância Mínima não apresentou nenhuma sensibilidade a elas, visto que este último não se alterou mesmo quando foi submetido a variações da média das amostras que por concepção do método seria o que mais influenciaria na discriminação das cores. Este fato pode ser importante quando se deseja discriminar outros tipos de variáveis nos dois primeiros métodos, visto que um estudo mais detalhado de suas amostras podem contribuir para um melhor entendimento dos resultados obtidos.

Apesar de se propor neste a comparação entre os métodos, não se pode deixar de lado algumas considerações observadas nos experimentos feitos.

O método de Máxima Verossimilhança, na sua forma original, só apresentou um desempenho razoável quando as médias e variância eram heterogêneas, ou seja, quando as componentes da cor não variaram igualmente, comportando-se de maneira aleatória. Isto foi realmente provado, pois ao selecionar-se amostras aleatórias na imagem digitalizada, obteve-se um bom resultado. Quando o método foi modificado, alterando-se as matrizes de covariância a fim de tornar as componentes da cor independentes, percebeu-se que o método funcionou perfeitamente, exceto na imagem digitalizada onde se notou que o índice de classificação correta (estatística *kappa*) sofreu um decréscimo. Esta diferença deve-se provavelmente ao tamanho da amostra por ser esta possivelmente muito pequena ou deve-se ainda à classificação visual feita na imagem, dado que este foi o único meio de obtê-la. Este método, portanto, não apresentou desempenho satisfatório no que se refere ao tempo de processamento, gastando cerca de 60% do tempo utilizado para os três métodos.

Em todos os experimentos, o método de Fisher, apresentou bom desempenho quando utilizado em sua forma original; e melhorou mais ainda quando se alterou as matrizes de covariância para que as componentes apresentassem independência entre si. Quanto ao tempo de processamento, o método de Fisher, gastou 30% do total do período despendido em todos os métodos, que considerando a complexidade de cálculos utilizados por ele, pode-se considerá-lo como razoavelmente rápido.

O desempenho do método de Distância Mínima foi excelente para qualquer um dos experimentos, apesar de utilizar um algoritmo simples e regra de decisão de fácil entendimento, o que possibilitou sua rapidez no tempo de processamento (10%).

Desta forma, conclui-se que o método de Distância Mínima é o mais apropriado quando se deseja discriminar cores reais em uma imagem, dado que este não sofreu alteração em nenhuma das amostras escolhidas, no número de cores discriminadas e na restrição de independência entre as componentes da cor. E, ainda, quando se necessita um reconhecimento em tempo real (*on line*), recomenda-se a sua utilização, mesmo que ao ser submetido a classificação da imagem digitalizada tivesse apresentado performance semelhante aos demais métodos.

Percebeu-se com estas análises que a variável “cor” é mais susceptível ao seu vetor médio e que outros aspectos como a variância e a distribuição a que mais se aproxima, não desempenharam o mesmo papel de importância. Assim, recomenda-se um estudo mais detalhado desta variável, utilizando-se, por exemplo, amostras reais com variações de iluminação.

Recomenda-se ainda estudar a discriminação de cores reais utilizando-se outras alternativas de representação, como no domínio espectral ou num outro sistema, por exemplo, o sistema IHS (*Intensity-Hue-Saturation*) descrito no capítulo 2.

Finalmente, percebeu-se no decorrer deste trabalho a importância de se conhecer a variável a ser discriminada quando se escolher um método estatístico, a fim de que esta não viole nenhuma hipótese concebida pelo método.

7. REFERÊNCIAS BIBLIOGRÁFICAS:

- ARGIALAS , Demetre P. & HALOW, Charles A. Computational Image Interpretation Models: An Overview and a Perspective. Photogrammetric Engineering and Remote Sensing. Vol 56, june 1990, pp. 871-886.
- BANON, Gerald J.F. Bases da Computação Gráfica. Ed. Campus, Rio de Janeiro, 1989.
- BOLSTAD, Paul V. & LILLESAND, T. M. Rapid Maximum Likelihood Classification. Photogrammetric Engineering & Remote Sensing. Vol. 57, January 1991, pp. 67-74.
- CHUVIECO, Emilio. Fundamentos de Teledetección Espacial. Ed. Ediciones Rialp, Madrid, 1990.
- COHEN, J. A Coefficient of Agreement for Nominal Scales. Educational and Psychological Measurement, vol. XX, No. 1, 1960
- COSTA NETO, Pedro L. O. Estatística. Ed. Edgard Blücher Ltda, São Paulo, 1977.
- DEUSEN, Paul C. Van. Modified Highest Confidence First Classification. Photogrammetric Engineering and Remote Sensing. Vol 61, No. 4, April 1995, pp. 419-425.
- DEVORE, Jay L. Probability and Statistics of Engineering and the Sciences. Brooks/Cole Publishing Company, Monterey-California, 1982.
- FOODY, Giles M. ; McCULLOCH, Mary B. & YATES, William B. Classification of Remotely Sensed Data by an Artificial Neural Network: Issues Related to Training Data Characteristics. Photogrammetric Engineering & Remote Sensing. Vol. 61, No.4, April 1995, pp 391-401.
- GARCIA, Gilberto J. Sensoriamento Remoto. Princípios e Interpretação de Imagens. Ed. Nobel, São Paulo, 1982.

- GREEN, William B. Digital Image Processing: A Systems Approach. Ed. Van Nostrand Reinhold, New York, 1989.
- GROOVER, Mikell P.; WEISS, Michell; NAGEL, Roger N. & ODREY, Nicholas G. Robótica. Tecnologia e Programação. Ed. McGraw-Hill. São Paulo, 1989.
- HAIR, Joseph F; ANDERSON, Rolph E.; TATHAM, Ronald L. & BLACK, William C. Multivariate Data Analysis. Third Edition. Ed. Macmillan Publishing Company, New York, 1987.
- HUANG, Kal-Yi & MAUSAL, P. M. Comparing a Piecewise Linear Classifier with Gaussian Maximum-Likelihood and Parallelepiped Classifiers in Terms of Accuracy and Speed. Photogrammetric Engineering and Remote Sensing. Vol 60, No. 11, Novembro 1994, pp. 1333-1338.
- JOHNSON, Richard A. & WICHERN, Dean W. Applied Multivariate Statistical Analysis. Ed. Prentice Hall International, 1985.
- LILLESAND, Thomas M & KIEFER, Ralph W. Remote Sensing and Image Interpretation. Ed. John Wiley & Sons, New York, 1994.
- MARDIA, K. V.; KENT J. T. & BIBBY, J.M. Multivariate Analysis. Academic Press, Inc, Orlando-Florida, 1982.
- MASCARENHAS, Nelson D. A. & VELASCO, Flávio R. D. Processamento Digital de Imagens. INPE/CNPQ. São Paulo, 1984.
- MATLAB User's Guide. The MathWorks, Inc. Bausch - Gall GmbH, 1990.
- MOIK, Johannes G. Digital Processing of Remotely Sensed Images. Goddard Space Flight Center. Ed. Library of Congress Cataloging in Publication Data, Washington D.C. , 1980.
- NOVO, E.M.L.M. Sensoramento Remoto. Princípios e Aplicações. Ed. Edgard Blücher Ltda. São Paulo, 1992.

- PEDDLE, Derek R. Knowledge Formulation for Supervised Evidential Classification. Photogrammetric Engineering and Remote Sensing. Vol 61, No. 4, April 1995, pp. 409-417.
- RICHARDS, John A. Remote Sensing Digital Image Analysis. Ed. Springer-Verlag Berlin Heidelberg, Alemanham 1986.
- SONKA, Milan; HLAVAC, Vaclav & BOYLE, Roger. Image Processing, Analysis and Machine Vision. Ed. Chapman & Hall, Cambridge, 1993.
- THALMANN, Nadia M. & THALMANN, Daniel. Image Synthesis. Teory and Practice. Ed. Springer-Velag, Tokio, 1987.
- THOMPSON, Clay M. & SHURE, Loren. Image Processing Toolbox User's Guide. The MathWorks, Inc., 1995.
- VAN DEUSEN, Paul C. Modified Highest Confidence First Classification. Photogrammetric Engineering & Remote Sensing. Vol. 61. April 1995, pp. 419-425.
- YOUNG, Tzay Y. & FU, King-Sun. Handbook of Pattern Recognition and Image Processing. Ed. Academic Press. Califórnia, 1986.
- ZHUANG, Xin; ENGEL, Bernard A.; XIONG, Xiaoping & JOHANNSEN, Chris J. Analysis of Classification Results of Remotely Sensed Data and Evaluation of Classification Algorithms. Photogrammetric Engineering & Remote Sensing. Vol. 61, No. 4, April 1995, pp 427-433.

ANEXO I

“Algoritmo de Comparação dos Métodos: Máxima Verossimilhança, Distância Mínima e de Fisher, para Reconhecimento de Cores no Sistema RGB em Imagens”.

Rotina: “algodis”

```
MP=menu('DISCRIMINAÇÃO DE CORES','ENTRADA DAS CORES', ...  
        'ANÁLISE DISCRIMINANTE', 'COMPARAÇÃO DOS MÉTODOS',...  
        'ANÁLISE VISUAL', 'TERMINAR')  
if MP==1;  
    entrecor  
    algodis  
elseif MP==2;  
    discrim  
    algodis  
elseif MP==3;  
    compara  
    algodis  
elseif MP==4;  
    show  
    algodis  
elseif MP==5;  
    termin  
end
```

Subrotina: “entrecor”

```
MCO=menu('ENTRADA DAS CORES','TIPO DE AMOSTRA','No. DE CORES')
  if MCO==1
    selmap
    entrecor
  elseif MCO==2
    selcor
  end
```

Subrotina:”discrim”

```
DIS=menu('ANÁLISE DISCRIMINANTE','MÉDIA E VARIÂNCIA',...
  'MÁXIMA VEROSSIMILHANÇA','DISTÂNCIA MÍNIMA',...
  'FISHER')
  if DIS==1
    medvar
    discrim
  elseif DIS==2
    maxver
    discrim
  elseif DIS==3
    distmin
    discrim
  elseif DIS==4
    fisher
  end
```

Subrotina: "compara"

```

TCM=zeros(N,N);
TCF=zeros(N,N);
TCD=zeros(N,N);
for i=1:m
    for j=1:N
        if xx(i)==j
            if ym(i)==j
                TCM(j,j)=TCM(j,j)+1;
            else
                k=ym(i);
                TCM(j,k)=TCM(j,k)+1;
            end
        if yf(i)==j
            TCF(j,j)=TCF(j,j)+1;
        else
            k=yf(i);
            TCF(j,k)=TCF(j,k)+1;
        end
        if ydm(i)==j
            TCD(j,j)=TCD(j,j)+1;
        else
            k=ydm(i);
            TCD(j,k)=TCD(j,k)+1;
        end
    end
end
end
end
end
end
for i=1:N
    sud(i)=sum(TCM(i,i));
    sul(i)=sum(TCM(i,:));
    suc(i)=sum(TCM(:,i));
end
PoM=sum(sud)/m;
PeM=(sul*suc')/(m^2);
kappa1=(PoM-PeM)/(1-PeM);
for i=1:N
    sud(i)=sum(TCF(i,i));
    sul(i)=sum(TCF(i,:));
    suc(i)=sum(TCF(:,i));
end
end

```

```

PoF=sum(sud)/m;
PeF=(sul*suc')/(m^2);
kappa2=(PoF-PeF)/(1-PeF);
for i=1:N
    sud(i)=sum(TCD(i,i));
    sul(i)=sum(TCD(i,:));
    suc(i)=sum(TCD(:,i));
end
PoD=sum(sud)/m;
PeD=(sul*suc')/(m^2);
kappa3=(PoD-PeD)/(1-PeD);
var1=(PoM*(1-PoM))/(m*((1-PeM)^2));
var2=(PoF*(1-PoF))/(m*((1-PeF)^2));
var3=(PoD*(1-PoD))/(m*((1-PeD)^2));
Zcfm=(kappa1-kappa2)/sqrt((var1+var2));
Zcfd=(kappa3-kappa2)/sqrt((var3+var2));
Zcmd=(kappa1-kappa3)/sqrt((var1+var3));
Zt1=norminv(0.025,0,1);
Zt2=norminv(0.975,0,1);
if Zcfm<=Zt1
    disp('EXISTE DIFERENÇA ENTRE MAXVER E FISHER')
elseif Zcfm>=Zt2
    disp('EXISTE DIFERENÇA ENTRE MAXVER E FISHER')
else
    disp('NÃO EXITE DIFERENÇA ENTRE MAXVER E FISHER')
end
if Zcfd<=Zt1
    disp('EXISTE DIFERENÇA ENTRE FISHER E DISTÂNCIA')
elseif Zcfd>=Zt2
    disp('EXISTE DIFERENÇA ENTRE FISHER E DISTÂNCIA')
else
    disp('NÃO EXITE DIFERENÇA ENTRE FISHER E DISTÂNCIA')
end
if Zcmd<=Zt1
    disp('EXISTE DIFERENÇA ENTRE MAXVER E DISTÂNCIA')
elseif Zcmd>=Zt2
    disp('EXISTE DIFERENÇA ENTRE MAXVER E DISTÂNCIA')
else
    disp('NÃO EXITE DIFERENÇA ENTRE MAXVER E DISTÂNCIA')
end

```

Subrotina: "show"

```

subplot(1,4,1)
imshow(R,G,B)

```

```

subplot(1,4,2)
imshow(RR,GG,BB)
subplot(1,4,3)
imshow(RRR,GGG,BBB)
subplot(1,4,4)
imshow(RRRR,GGGG,BBBB)

```

Subrotina: "termin"

```

de1=menu('DESEJA SAIR DO PROGRAMA?','SIM, SALVANDO VARIÁVEIS',...
        'SIM, SEM SALVAR VARIÁVEIS','NÃO, SALVANDO VARIÁVEIS',...
        'NÃO, SEM SALVAR VARIÁVEIS')
if de1==1
    save discri.dat ym, yf, TCM, TCF, kappa1, kappa2, Zc, dtm, dtf
elseif de1==2
    disp('FIM')
elseif de1==3
    save discri.dat ym, yf, TCM, TCF, kappa1, kappa2, Zc, dtm, dtf
    algodis
elseif de1==4
    algodis
end

```

Subrotina: "selmap"

```

M=menu('TIPO DE AMOSTRA:','MÉDIA PROX. E VARIÂNCIA PEQ.','MÉDIA
AFAST. E VARIÂNCIA PEQ.',...
      'VARIÂNCIA GRANDE','MÉDIAS DIFERENTES','VARIÂNCIA DIFERENTE');
if M==1
    mapa1
elseif M==2
    mapa2
elseif M==3
    mapa3
elseif M==4
    mapa4
elseif M==5
    mapa5
end

```

Subrotina: "selcor"

```

MS1=menu('No. CORES DISCRIMINADAS','2 CORES','3 CORES',...
        '5 CORES','6 CORES', '10 CORES', '15 CORES')
if MS1==1

```

```

disp('Selecione as 2 Cores:')
c=input('ENTRE COM A COR 1:');
cor
X1=X;
Y1=Y;
disp('Cor 2:');
c=input('ENTRE COM A COR 2:');
cor
X2=X;
Y2=Y;
N=2;
corr2
elseif MS1==2
disp('Selecione as 3 cores');
c=input('ENTRE COM A COR 1:');
cor
X1=X;
Y1=Y;
c=input('ENTRE COM A COR 2:');
cor
X2=X;
Y2=Y;
c=input('ENTRE COM A COR 3:');
cor
X3=X;
Y3=Y;
N=3;
corr3
elseif MS1==3
disp('Selecione as 5 cores');
c=input('ENTRE COM A COR 1:');
cor
X1=X;
Y1=Y;
c=input('ENTRE COM A COR 2:');
cor
X2=X;
Y2=Y;
c=input('ENTRE COM A COR 3:');
cor
X3=X;
Y3=Y;
c=input('ENTRE COM A COR 4:');
cor
X4=X;

```

```

Y4=Y;
c=input('ENTRE COM A COR 5:');
cor
X5=X;
Y5=Y;
N=5;
corr5
elseif MS1==4
disp('Selecione as 6 cores');
c=input('ENTRE COM A COR 1:');
cor
X1=X;
Y1=Y;
c=input('ENTRE COM A COR 2:');
cor
X2=X;
Y2=Y;
c=input('ENTRE COM A COR 3:');
cor
X3=X;
Y3=Y;
c=input('ENTRE COM A COR 4:');
cor
X4=X;
Y4=Y;
c=input('ENTRE COM A COR 5:');
cor
X5=X;
Y5=Y;
N=6;
corr6
elseif MS1==5
disp('Selecione as 10 cores');
c=input('ENTRE COM A COR 1:');
cor
X1=X;
Y1=Y;
c=input('ENTRE COM A COR 2:');
cor
X2=X;
Y2=Y;
c=input('ENTRE COM A COR 3:');
cor
X3=X;
Y3=Y;

```

```
c=input('ENTRE COM A COR 4:');
cor
X4=X;
Y4=Y;
c=input('ENTRE COM A COR 5:');
cor
X5=X;
Y5=Y;
c=input('ENTRE COM A COR 6:');
cor
X6=X;
Y6=Y;
c=input('ENTRE COM A COR 7:');
cor
X7=X;
Y7=Y;
c=input('ENTRE COM A COR 8:');
cor
X8=X;
Y8=Y;
c=input('ENTRE COM A COR 9:');
cor
X9=X;
Y9=Y;
c=input('ENTRE COM A COR 10:');
cor
X10=X;
Y10=Y;
N=10;
corr10
elseif MS1==6
disp('Selecione as 15 cores');
c=input('ENTRE COM A COR 1:');
cor
X1=X;
Y1=Y;
c=input('ENTRE COM A COR 2:');
cor
X2=X;
Y2=Y;
c=input('ENTRE COM A COR 3:');
cor
X3=X;
Y3=Y;
c=input('ENTRE COM A COR 4:');
```

```
cor
X4=X;
Y4=Y;
c=input('ENTRE COM A COR 5:');
cor
X5=X;
Y5=Y;
c=input('ENTRE COM A COR 6:');
cor
X6=X;
Y6=Y;
c=input('ENTRE COM A COR 7:');
cor
X7=X;
Y7=Y;
c=input('ENTRE COM A COR 8:');
cor
X8=X;
Y8=Y;
c=input('ENTRE COM A COR 9:');
cor
X9=X;
Y9=Y;
c=input('ENTRE COM A COR 10:');
cor
X10=X;
Y10=Y;
c=input('ENTRE COM A COR 11:');
cor
X11=X;
Y11=Y;
c=input('ENTRE COM A COR 12:');
cor
X12=X;
Y12=Y;
c=input('ENTRE COM A COR 13:');
cor
X13=X;
Y13=Y;
c=input('ENTRE COM A COR 14:');
cor
X14=X;
Y14=Y;
c=input('ENTRE COM A COR 15:');
cor
```

```

X15=X;
Y15=Y;
N=15;
corr15
end

```

Subrotina: "medvar"

```

if N==2
[m,n]=size(x);
MX1=mean(X1);
MX2=mean(X2);
[m1,n1]=size(X1);
[m2,n2]=size(X2);
S1=cov(X1);
S2=cov(X2);
elseif N==3
[m,n]=size(x);
MX1=mean(X1);
MX2=mean(X2);
MX3=mean(X3);
[m1,n1]=size(X1);
[m2,n2]=size(X2);
[m3,n3]=size(X3);
S1=cov(X1);
S2=cov(X2);
S3=cov(X3);
elseif N==5
[m,n]=size(x);
MX1=mean(X1);
MX2=mean(X2);
MX3=mean(X3);
MX4=mean(X4);
MX5=mean(X5);
[m1,n1]=size(X1);
[m2,n2]=size(X2);
[m3,n3]=size(X3);
[m4,n4]=size(X4);
[m5,n5]=size(X5);
S1=cov(X1);
S2=cov(X2);
S3=cov(X3);

```

```
S4=cov(X4);
S5=cov(X5);
elseif N==6
[m,n]=size(x);
MX1=mean(X1);
MX2=mean(X2);
MX3=mean(X3);
MX4=mean(X4);
MX5=mean(X5);
MX6=mean(X6);
[m1,n1]=size(X1);
[m2,n2]=size(X2);
[m3,n3]=size(X3);
[m4,n4]=size(X4);
[m5,n5]=size(X5);
[m6,n6]=size(X6);
S1=cov(X1);
S2=cov(X2);
S3=cov(X3);
S4=cov(X4);
S5=cov(X5);
S6=cov(X6);
elseif N==10
[m,n]=size(x);
MX1=mean(X1);
MX2=mean(X2);
MX3=mean(X3);
MX4=mean(X4);
MX5=mean(X5);
MX6=mean(X6);
MX7=mean(X7);
MX8=mean(X8);
MX9=mean(X9);
MX10=mean(X10);
[m1,n1]=size(X1);
[m2,n2]=size(X2);
[m3,n3]=size(X3);
[m4,n4]=size(X4);
[m5,n5]=size(X5);
[m6,n6]=size(X6);
[m7,n7]=size(X7);
[m8,n8]=size(X8);
[m9,n9]=size(X9);
[m10,n10]=size(X10);
S1=cov(X1);
```

```
S2=cov(X2);
S3=cov(X3);
S4=cov(X4);
S5=cov(X5);
S6=cov(X6);
S7=cov(X7);
S8=cov(X8);
S9=cov(X9);
S10=cov(X10);
elseif N==15
[m,n]=size(x);
MX1=mean(X1);
MX2=mean(X2);
MX3=mean(X3);
MX4=mean(X4);
MX5=mean(X5);
MX6=mean(X6);
MX7=mean(X7);
MX8=mean(X8);
MX9=mean(X9);
MX10=mean(X10);
MX11=mean(X11);
MX12=mean(X12);
MX13=mean(X13);
MX14=mean(X14);
MX15=mean(X15);
[m1,n1]=size(X1);
[m2,n2]=size(X2);
[m3,n3]=size(X3);
[m4,n4]=size(X4);
[m5,n5]=size(X5);
[m6,n6]=size(X6);
[m7,n7]=size(X7);
[m8,n8]=size(X8);
[m9,n9]=size(X9);
[m10,n10]=size(X10);
[m11,n11]=size(X11);
[m12,n12]=size(X12);
[m13,n13]=size(X13);
[m14,n14]=size(X14);
[m15,n15]=size(X15);
S1=cov(X1);
S2=cov(X2);
S3=cov(X3);
S4=cov(X4);
```

```

S5=cov(X5);
S6=cov(X6);
S7=cov(X7);
S8=cov(X8);
S9=cov(X9);
S10=cov(X10);
S11=cov(X11);
S12=cov(X12);
S13=cov(X13);
S14=cov(X14);
S15=cov(X15);
end

```

Subrotina: "maxver"

```

if N==2
t1m=clock;
for i=1:m
g1=-(log(abs(S1)))-(x(i,:) - MX1)*(pinv(S1))*(x(i,:) - MX1)';
g2=-(log(abs(S2)))-(x(i,:) - MX2)*(pinv(S2))*(x(i,:) - MX2)';
if g1>g2
ym(i,:)=1;
y(i,:)=MX1;
else
ym(i,:)=2;
y(i,:)=MX2;
end
end
t2m=clock;
dtm=etime(t2m,t1m);
elseif N==3
t1m=clock;
for i=1:m
g1=-(log(abs(S1)))-(x(i,:) - MX1)*(pinv(S1))*(x(i,:) - MX1)';
g2=-(log(abs(S2)))-(x(i,:) - MX2)*(pinv(S2))*(x(i,:) - MX2)';
g3=-(log(abs(S3)))-(x(i,:) - MX3)*(pinv(S3))*(x(i,:) - MX3)';
g11=max(max(g1));
g22=max(max(g2));
g33=max(max(g3));
g=max([g11 g22 g33]);
if g==g11
ym(i,:)=1;
y(i,:)=Y(1,:);
else

```

```

end
if g==g22
    ym(i,:)=2;
    y(i,:)=Y(2,:);
else
end
if g==g33
    ym(i,:)=3;
    y(i,:)=Y(3,:);
else
end
end
t2m=clock;
dtm=etime(t2m,t1m);
elseif N==5
t1m=clock;
for i=1:m
    g1=-(log(abs(S1)))-(x(i,:) - MX1)'*(pinv(S1))*(x(i,:) - MX1);
    g2=-(log(abs(S2)))-(x(i,:) - MX2)'*(pinv(S2))*(x(i,:) - MX2);
    g3=-(log(abs(S3)))-(x(i,:) - MX3)'*(pinv(S3))*(x(i,:) - MX3);
    g4=-(log(abs(S4)))-(x(i,:) - MX4)'*(pinv(S4))*(x(i,:) - MX4);
    g5=-(log(abs(S5)))-(x(i,:) - MX5)'*(pinv(S5))*(x(i,:) - MX5);
    g11=max(max(g1));
    g22=max(max(g2));
    g33=max(max(g3));
    g44=max(max(g4));
    g55=max(max(g5));
    g=max([g11 g22 g33 g44 g55]);
    if g==g11
        ym(i,:)=1;
        y(i,:)=Y(1,:);
    else
    end
    if g==g22
        ym(i,:)=2;
        y(i,:)=Y(2,:);
    else
    end
    if g==g33
        ym(i,:)=3;
        y(i,:)=Y(3,:);
    else
    end
    if g==g44
        ym(i,:)=4;

```

```

    y(i,:)=Y(4,:);
else
end
if g==g55
    ym(i,:)=5;
    y(i,:)=Y(5,:);
else
end
end
t2m=clock;
dtm=etime(t2m,t1m);
elseif N==6
t1m=clock;
for i=1:m
    g1=-(log(det(S1)))-(x(i,:)' - MX1')*(pinv(S1))*(x(i,:)' - MX1');
    g2=-(log(det(S2)))-(x(i,:)' - MX2')*(pinv(S2))*(x(i,:)' - MX2');
    g3=-(log(det(S3)))-(x(i,:)' - MX3')*(pinv(S3))*(x(i,:)' - MX3');
    g4=-(log(det(S4)))-(x(i,:)' - MX4')*(pinv(S4))*(x(i,:)' - MX4');
    g5=-(log(det(S5)))-(x(i,:)' - MX5')*(pinv(S5))*(x(i,:)' - MX5');
    g6=-(log(det(S6)))-(x(i,:)' - MX6')*(pinv(S6))*(x(i,:)' - MX6');
    g=max([g1 g2 g3 g4 g5 g6]);
    if g==g1
        ym(i,:)=1;
        y(i,:)=Y(1,:);
    else
    end
    if g==g2
        ym(i,:)=2;
        y(i,:)=Y(2,:);
    else
    end
    if g==g3
        ym(i,:)=3;
        y(i,:)=Y(3,:);
    else
    end
    if g==g4
        ym(i,:)=4;
        y(i,:)=Y(4,:);
    else
    end
    if g==g5
        ym(i,:)=5;
        y(i,:)=Y(5,:);
    else

```

```

end
if g==g6
    ym(i,:)=6;
    y(i,:)=Y(6,:);
else
end
end
t2m=clock;
dtm=etime(t2m,t1m);
elseif N==10
t1m=clock;
for i=1:m
    g1=-(log(abs(S1)))-(x(i,:)' - MX1')*(pinv(S1))*(x(i,:)' - MX1');
    g2=-(log(abs(S2)))-(x(i,:)' - MX2')*(pinv(S2))*(x(i,:)' - MX2');
    g3=-(log(abs(S3)))-(x(i,:)' - MX3')*(pinv(S3))*(x(i,:)' - MX3');
    g4=-(log(abs(S4)))-(x(i,:)' - MX4')*(pinv(S4))*(x(i,:)' - MX4');
    g5=-(log(abs(S5)))-(x(i,:)' - MX5')*(pinv(S5))*(x(i,:)' - MX5');
    g6=-(log(abs(S6)))-(x(i,:)' - MX6')*(pinv(S6))*(x(i,:)' - MX6');
    g7=-(log(abs(S7)))-(x(i,:)' - MX7')*(pinv(S7))*(x(i,:)' - MX7');
    g8=-(log(abs(S8)))-(x(i,:)' - MX8')*(pinv(S8))*(x(i,:)' - MX8');
    g9=-(log(abs(S9)))-(x(i,:)' - MX9')*(pinv(S9))*(x(i,:)' - MX9');
    g10=-(log(abs(S10)))-(x(i,:)' - MX10')*(pinv(S10))*(x(i,:)' - MX10');
    g11=max(max(g1));
    g22=max(max(g2));
    g33=max(max(g3));
    g44=max(max(g4));
    g55=max(max(g5));
    g66=max(max(g6));
    g77=max(max(g7));
    g88=max(max(g8));
    g99=max(max(g9));
    g1010=max(max(g10));
    g=max([g11 g22 g33 g44 g55 g66 g77 g88 g99 g1010]);
    if g==g11
        ym(i,:)=1;
        y(i,:)=Y(1,:);
    else
    end
    if g==g22
        ym(i,:)=2;
        y(i,:)=Y(2,:);
    else
    end
    if g==g33
        ym(i,:)=3;

```

```

    y(i,:)=Y(3,:);
else
end
if g==g44
    ym(i,:)=4;
    y(i,:)=Y(4,:);
else
end
if g==g55
    ym(i,:)=5;
    y(i,:)=Y(5,:);
else
end
if g==g66
    ym(i,:)=6;
    y(i,:)=Y(6,:);
else
end
if g==g77
    ym(i,:)=7;
    y(i,:)=Y(7,:);
else
end
if g==g88
    ym(i,:)=8;
    y(i,:)=Y(8,:);
else
end
if g==g99
    ym(i,:)=9;
    y(i,:)=Y(9,:);
else
end
if g==g1010
    ym(i,:)=10;
    y(i,:)=Y(10,:);
else
end
end
t2m=clock;
dtm=etime(t2m,t1m);
elseif N==15
t1m=clock;
for i=1:m
    g1=-((log(det(S1)))-(x(i,:)' - MX1')*(pinv(S1))*(x(i,:)' - MX1'));

```

```

g2=- (log(det(S2)))-(x(i,:) - MX2')*(pinv(S2))*(x(i,:) - MX2');
g3=- (log(det(S3)))-(x(i,:) - MX3')*(pinv(S3))*(x(i,:) - MX3');
g4=- (log(det(S4)))-(x(i,:) - MX4')*(pinv(S4))*(x(i,:) - MX4');
g5=- (log(det(S5)))-(x(i,:) - MX5')*(pinv(S5))*(x(i,:) - MX5');
g6=- (log(det(S6)))-(x(i,:) - MX6')*(pinv(S6))*(x(i,:) - MX6');
g7=- (log(det(S7)))-(x(i,:) - MX7')*(pinv(S7))*(x(i,:) - MX7');
g8=- (log(det(S8)))-(x(i,:) - MX8')*(pinv(S8))*(x(i,:) - MX8');
g9=- (log(det(S9)))-(x(i,:) - MX9')*(pinv(S9))*(x(i,:) - MX9');
g10=- (log(det(S10)))-(x(i,:) - MX10')*(pinv(S10))*(x(i,:) - MX10');
g11=- (log(det(S11)))-(x(i,:) - MX11')*(pinv(S11))*(x(i,:) - MX11');
g12=- (log(det(S12)))-(x(i,:) - MX12')*(pinv(S12))*(x(i,:) - MX12');
g13=- (log(det(S13)))-(x(i,:) - MX13')*(pinv(S13))*(x(i,:) - MX13');
g14=- (log(det(S14)))-(x(i,:) - MX14')*(pinv(S14))*(x(i,:) - MX14');
g15=- (log(det(S15)))-(x(i,:) - MX15')*(pinv(S15))*(x(i,:) - MX15');
g=max([g1 g2 g3 g4 g5 g6 g7 g8 g9 g10 g11 g12 g13 g14 g15 ]);
if g==g1
    ym(i,:)=1;
    y(i,:)=Y(6,:);
else
end
if g==g2
    ym(i,:)=2;
    y(i,:)=Y(6,:);
else
end
if g==g3
    ym(i,:)=3;
    y(i,:)=Y(6,:);
else
end
if g==g4
    ym(i,:)=4;
    y(i,:)=Y(6,:);
else
end
if g==g5
    ym(i,:)=5;
    y(i,:)=Y(6,:);
else
end
if g==g6
    ym(i,:)=6;
    y(i,:)=Y(6,:);
else
end

```

```
if g==g7
    ym(i,:)=7;
    y(i,:)=Y(13,:);
else
end
if g==g8
    ym(i,:)=8;
    y(i,:)=Y(13,:);
else
end
if g==g9
    ym(i,:)=9;
    y(i,:)=Y(13,:);
else
end
if g==g10
    ym(i,:)=10;
    y(i,:)=Y(12,:);
else
end
if g==g11
    ym(i,:)=11;
    y(i,:)=Y(13,:);
else
end
if g==g12
    ym(i,:)=12;
    y(i,:)=Y(13,:);
else
end
if g==g13
    ym(i,:)=13;
    y(i,:)=Y(13,:);
else
end
if g==g14
    ym(i,:)=14;
    y(i,:)=Y(13,:);
else
end
if g==g15
    ym(i,:)=15;
    y(i,:)=Y(13,:);
else
end
```

```

end
t2m=clock;
dtm=etime(t2m,t1m);
end
AM=[y(:,1) y(:,2) y(:,3)];
A1=(AM(:,1))';
A2=(AM(:,2))';
A3=(AM(:,3))';
[h1,h2]=size(R);
for j=1:h2
    k1=(j-1)*h1;
    for i=1:h1
        k2=k1+i;
        RR(i,j)=A1(1,k2);
        GG(i,j)=A2(1,k2);
        BB(i,j)=A3(1,k2);
    end
end

```

```

end

```

Subrotina: "distmin"

```

if N==2
    t1d=clock;
    for i=1:m
        DM1=(x(i,:)-MX1)'*(x(i,:)-MX1);
        DM2=(x(i,:)-MX2)'*(x(i,:)-MX2);
        if DM1<DM2
            ydm(i,:)=1;
            YDM(i,:)=MX1;
        else
            ydm(i,:)=2;
            YDM(i,:)=MX2;
        end
    end
end
t2d=clock;
dtd=etime(t2d,t1d);
AM=[YDM(:,1) YDM(:,2) YDM(:,3)];
A1=(AM(:,1))';
A2=(AM(:,2))';
A3=(AM(:,3))';
[h1,h2]=size(R);
for j=1:h2
    k1=(j-1)*h1;
    for i=1:h1

```

```

        k2=k1+i;
        RRRR(i,j)=A1(1,k2);
        GGGG(i,j)=A2(1,k2);
        BBBB(i,j)=A3(1,k2);
    end
end
elseif N==3
t1d=clock;
for i=1:m
    DM1=(x(i,:)-MX1)'.*(x(i,:)-MX1);
    DM2=(x(i,:)-MX2)'.*(x(i,:)-MX2);
    DM3=(x(i,:)-MX3)'.*(x(i,:)-MX3);
    DM=min([DM1 DM2 DM3]);
    if DM==DM1
        ydm(i,:)=1;
        YDM(i,:)=MX1;
    else
    end
    if DM==DM2
        ydm(i,:)=2;
        YDM(i,:)=MX2;
    else
    end
    if DM==DM3
        ydm(i,:)=3;
        YDM(i,:)=MX3;
    else
    end
end
t2d=clock;
dtd=etime(t2d,t1d);
AM=[YDM(:,1) YDM(:,2) YDM(:,3)];
A1=(AM(:,1))';
A2=(AM(:,2))';
A3=(AM(:,3))';
[h1,h2]=size(R);
for j=1:h2
    k1=(j-1)*h1;
    for i=1:h1
        k2=k1+i;
        RRRR(i,j)=A1(1,k2);
        GGGG(i,j)=A2(1,k2);
        BBBB(i,j)=A3(1,k2);
    end
end
end

```

```

elseif N==5
t1d=clock;
for i=1:m
    DM1=(x(i,:)-MX1')*(x(i,:)-MX1');
    DM2=(x(i,:)-MX2')*(x(i,:)-MX2');
    DM3=(x(i,:)-MX3')*(x(i,:)-MX3');
    DM4=(x(i,:)-MX4')*(x(i,:)-MX4');
    DM5=(x(i,:)-MX5')*(x(i,:)-MX5');
DM=min([DM1 DM2 DM3 DM4 DM5]);
if DM==DM1
    ydm(i,:)=1;
    YDM(i,:)=MX1;
else
end
if DM==DM2
    ydm(i,:)=2;
    YDM(i,:)=MX2;
else
end
if DM==DM3
    ydm(i,:)=3;
    YDM(i,:)=MX3;
else
end
if DM==DM4
    ydm(i,:)=4;
    YDM(i,:)=MX4;
else
end
if DM==DM5
    ydm(i,:)=5;
    YDM(i,:)=MX5;
else
end
end
t2d=clock;
dtd=etime(t2d,t1d);
AM=[YDM(:,1) YDM(:,2) YDM(:,3)];
A1=(AM(:,1))';
A2=(AM(:,2))';
A3=(AM(:,3))';
[h1,h2]=size(R);
for j=1:h2
    k1=(j-1)*h1;
    for i=1:h1

```

```

    k2=k1+i;
    RRRR(i,j)=A1(1,k2);
    GGGG(i,j)=A2(1,k2);
    BBBB(i,j)=A3(1,k2);
end
end
elseif N==6
    tld=clock;
    for i=1:m
        DM1=(x(i,:)-MX1)'*(x(i,:)-MX1);
        DM2=(x(i,:)-MX2)'*(x(i,:)-MX2);
        DM3=(x(i,:)-MX3)'*(x(i,:)-MX3);
        DM4=(x(i,:)-MX4)'*(x(i,:)-MX4);
        DM5=(x(i,:)-MX5)'*(x(i,:)-MX5);
        DM6=(x(i,:)-MX6)'*(x(i,:)-MX6);
        DM=min([DM1 DM2 DM3 DM4 DM5 DM6]);
        if DM==DM1
            ydm(i,:)=1;
            YDM(i,:)=MX1;
        else
            end
        if DM==DM2
            ydm(i,:)=2;
            YDM(i,:)=MX2;
        else
            end
        if DM==DM3
            ydm(i,:)=3;
            YDM(i,:)=MX3;
        else
            end
        if DM==DM4
            ydm(i,:)=4;
            YDM(i,:)=MX4;
        else
            end
        if DM==DM5
            ydm(i,:)=5;
            YDM(i,:)=MX5;
        else
            end
        if DM==DM6
            ydm(i,:)=6;
            YDM(i,:)=MX6;
        else

```

```

end
end
t2d=clock;
dtd=etime(t2d,t1d);
AM=[YDM(:,1) YDM(:,2) YDM(:,3)];
A1=(AM(:,1))';
A2=(AM(:,2))';
A3=(AM(:,3))';
[h1,h2]=size(R);
for j=1:h2
    k1=(j-1)*h1;
    for i=1:h1
        k2=k1+i;
        RRRR(i,j)=A1(1,k2);
        GGGG(i,j)=A2(1,k2);
        BBBB(i,j)=A3(1,k2);
    end
end
elseif N==10
t1d=clock;
for i=1:m
    DM1=(x(i,:)-MX1')*(x(i,:)-MX1');
    DM2=(x(i,:)-MX2')*(x(i,:)-MX2');
    DM3=(x(i,:)-MX3')*(x(i,:)-MX3');
    DM4=(x(i,:)-MX4')*(x(i,:)-MX4');
    DM5=(x(i,:)-MX5')*(x(i,:)-MX5');
    DM6=(x(i,:)-MX6')*(x(i,:)-MX6');
    DM7=(x(i,:)-MX7')*(x(i,:)-MX7');
    DM8=(x(i,:)-MX8')*(x(i,:)-MX8');
    DM9=(x(i,:)-MX9')*(x(i,:)-MX9');
    DM10=(x(i,:)-MX10')*(x(i,:)-MX10');
DM=min([DM1 DM2 DM3 DM4 DM5 DM6 DM7 DM8 DM9 DM10]);
if DM==DM1
    ydm(i,:)=1;
    YDM(i,:)=MX1;
else
end
if DM==DM2
    ydm(i,:)=2;
    YDM(i,:)=MX2;
else
end
if DM==DM3
    ydm(i,:)=3;
    YDM(i,:)=MX3;

```

```

else
end
if DM==DM4
    ydm(i,:)=4;
    YDM(i,:)=MX4;
else
end
if DM==DM5
    ydm(i,:)=5;
    YDM(i,:)=MX5;
else
end
if DM==DM6
    ydm(i,:)=6;
    YDM(i,:)=MX6;
else
end
if DM==DM7
    ydm(i,:)=7;
    YDM(i,:)=MX7;
else
end
if DM==DM8
    ydm(i,:)=8;
    YDM(i,:)=MX8;
else
end
if DM==DM9
    ydm(i,:)=9;
    YDM(i,:)=MX9;
else
end
if DM==DM10
    ydm(i,:)=10;
    YDM(i,:)=MX10;
else
end
end
t2d=clock;
dtd=etime(t2d,t1d);
AM=[YDM(:,1) YDM(:,2) YDM(:,3)];
A1=(AM(:,1))';
A2=(AM(:,2))';
A3=(AM(:,3))';
[h1,h2]=size(R);

```

```

for j=1:h2
    k1=(j-1)*h1;
    for i=1:h1
        k2=k1+i;
        RRRR(i,j)=A1(1,k2);
        GGGG(i,j)=A2(1,k2);
        BBBB(i,j)=A3(1,k2);
    end
end
elseif N==15
t1d=clock;
for i=1:m
    DM1=(x(i,:)-MX1')*(x(i,:)-MX1');
    DM2=(x(i,:)-MX2')*(x(i,:)-MX2');
    DM3=(x(i,:)-MX3')*(x(i,:)-MX3');
    DM4=(x(i,:)-MX4')*(x(i,:)-MX4');
    DM5=(x(i,:)-MX5')*(x(i,:)-MX5');
    DM6=(x(i,:)-MX6')*(x(i,:)-MX6');
    DM7=(x(i,:)-MX7')*(x(i,:)-MX7');
    DM8=(x(i,:)-MX8')*(x(i,:)-MX8');
    DM9=(x(i,:)-MX9')*(x(i,:)-MX9');
    DM10=(x(i,:)-MX10')*(x(i,:)-MX10');
    DM11=(x(i,:)-MX11')*(x(i,:)-MX11');
    DM12=(x(i,:)-MX12')*(x(i,:)-MX12');
    DM13=(x(i,:)-MX13')*(x(i,:)-MX13');
    DM14=(x(i,:)-MX14')*(x(i,:)-MX14');
    DM15=(x(i,:)-MX15')*(x(i,:)-MX15');
DM=min([DM1 DM2 DM3 DM4 DM5 DM6 DM7 DM8 DM9 DM10 DM11 DM12 DM13
DM14 DM15]);
    if DM==DM1
        ydm(i,:)=1;
        YDM(i,:)=Y(6,:);
    else
    end
    if DM==DM2
        ydm(i,:)=2;
        YDM(i,:)=Y(6,:);
    else
    end
    if DM==DM3
        ydm(i,:)=3;
        YDM(i,:)=Y(6,:);
    else
    end
    if DM==DM4

```

```
ydm(i,:)=4;
YDM(i,:)=Y(6,:);
else
end
if DM==DM5
ydm(i,:)=5;
YDM(i,:)=Y(6,:);
else
end
if DM==DM6
ydm(i,:)=6;
YDM(i,:)=Y(6,:);
else
end
if DM==DM7
ydm(i,:)=7;
YDM(i,:)=Y(13,:);
else
end
if DM==DM8
ydm(i,:)=8;
YDM(i,:)=Y(13,:);
else
end
if DM==DM9
ydm(i,:)=9;
YDM(i,:)=Y(13,:);
else
end
if DM==DM10
ydm(i,:)=10;
YDM(i,:)=Y(12,:);
else
end
if DM==DM11
ydm(i,:)=11;
YDM(i,:)=Y(13,:);
else
end
if DM==DM12
ydm(i,:)=12;
YDM(i,:)=Y(13,:);
else
end
if DM==DM13
```

```

    ydm(i,:)=13;
    YDM(i,:)=Y(13,:);
else
end
if DM==DM14
    ydm(i,:)=14;
    YDM(i,:)=Y(13,:);
else
end
if DM==DM15
    ydm(i,:)=15;
    YDM(i,:)=Y(13,:);
else
end
end
t2d=clock;
dtd=etime(t2d,t1d);
AM=[YDM(:,1) YDM(:,2) YDM(:,3)];
A1=(AM(:,1))';
A2=(AM(:,2))';
A3=(AM(:,3))';
[h1,h2]=size(R);
for j=1:h2
    k1=(j-1)*h1;
    for i=1:h1
        k2=k1+i;
        RRRR(i,j)=A1(1,k2);
        GGGG(i,j)=A2(1,k2);
        BBBB(i,j)=A3(1,k2);
    end
end
end
end

```

Subrotina: "fisher"

```

if N==2
    t1f=clock;
    MG=((m1*MX1)+(m2*MX2))*(1/(m1+m2));
    Sc((((n1-1)*S1)+((n2-1)*S2))/(n1+n2-2);
    W=(n1+n2-2)*Sc;
    Bo=((MX1-MG)*(MX1-MG))+ ((MX2-MG)*(MX2-MG));
    MX=[MX1;MX2];
elseif N==3
    t1f=clock;

```

```

MG=((m1*MX1)+(m2*MX2)+(m3*MX3))*(1/(m1+m2+m3));
Sc((((n1-1)*S1)+((n2-1)*S2)+((n3-1)*S3))/(n1+n2+n3-3);
W=(n1+n2+n3-3)*Sc;
Bo=((MX1-MG)*(MX1-MG))+((MX2-MG)*(MX2-MG))+((MX3-MG)*(MX3-MG));
MX=[MX1;MX2;MX3];
elseif N==5
t1f=clock;
MG=((m1*MX1)+(m2*MX2)+(m3*MX3)+(m4*MX4)+(m5*MX5))*(1/(m1+m2+m3+m4+m5));
Sc((((n1-1)*S1)+((n2-1)*S2)+((n3-1)*S3)+((n4-1)*S4)+((n5-1)*S5))/(n1+n2+n3+n4+n5-5);
W=(n1+n2+n3+n4+n5-5)*Sc;
Bo=((MX1-MG)*(MX1-MG))+((MX2-MG)*(MX2-MG))+((MX3-MG)*(MX3-MG))+((MX4-MG)*(MX4-MG))+((MX5-MG)*(MX5-MG));
MX=[MX1;MX2;MX3;MX4;MX5];
elseif N==6
t1f=clock;
MG=((m1*MX1)+(m2*MX2)+(m3*MX3)+(m4*MX4)+(m5*MX5)+(m6*MX6))*(1/(m1+m2+m3+m4+m5+m6));
Sc((((n1-1)*S1)+((n2-1)*S2)+((n3-1)*S3)+((n4-1)*S4)+((n5-1)*S5)+((n6-1)*S6))/(n1+n2+n3+n4+n5+n6-6);
W=(n1+n2+n3+n4+n5+n6-6)*Sc;
Bo=((MX1-MG)*(MX1-MG))+((MX2-MG)*(MX2-MG))+((MX3-MG)*(MX3-MG))+((MX4-MG)*(MX4-MG))+((MX5-MG)*(MX5-MG))+((MX6-MG)*(MX6-MG));
MX=[MX1;MX2;MX3;MX4;MX5;MX6];
elseif N==10
t1f=clock;
MG=((m1*MX1)+(m2*MX2)+(m3*MX3)+(m4*MX4)+(m5*MX5)+(m6*MX6)+(m7*MX7)+(m8*MX8)+(m9*MX9)+(m10*MX10))*(1/(m1+m2+m3+m4+m5+m6+m7+m8+m9+m10));
Sc(((n1-1)*S1+(n2-1)*S2+(n3-1)*S3+(n4-1)*S4+(n5-1)*S5+(n6-1)*S6+(n7-1)*S7+(n8-1)*S8+(n9-1)*S9+(n10-1)*S10)/(n1+n2+n3+n4+n5+n6+n7+n8+n9+n10-10);
W=(n1+n2+n3+n4+n5+n6+n7+n8+n9+n10-10)*Sc;
Bo=((MX1-MG)*(MX1-MG))+((MX2-MG)*(MX2-MG))+((MX3-MG)*(MX3-MG))+((MX4-MG)*(MX4-MG))+((MX5-MG)*(MX5-MG))+((MX6-MG)*(MX6-MG))+((MX7-MG)*(MX7-MG))+((MX8-MG)*(MX8-MG))+((MX9-MG)*(MX9-MG))+((MX10-MG)*(MX10-MG));
MX=[MX1;MX2;MX3;MX4;MX5;MX6;MX7;MX8;MX9;MX10];
elseif N==15
t1f=clock;
MG=((m1*MX1)+(m2*MX2)+(m3*MX3)+(m4*MX4)+(m5*MX5)+(m6*MX6)+(m7*MX7)+(m8*MX8)+(m9*MX9)+(m10*MX10)+(m11*MX11)+(m12*MX12)+(m13*MX13)+(m14*MX14)+(m15*MX15))*(1/(m1+m2+m3+m4+m5+m6+m7+m8+m9+m10+m11+m12+m13+m14+m15));

```

```

Sc=((n1-1)*S1+(n2-1)*S2+(n3-1)*S3+(n4-1)*S4+(n5-1)*S5+(n6-1)*S6+(n7-1)*S7+(n8-
1)*S8+(n9-1)*S9+(n10-1)*S10+(n11-1)*S11+(n12-1)*S12+(n13-1)*S13+(n14-
1)*S14+(n15-1)*S15)/(n1+n2+n3+n4+n5+n6+n7+n8+n9+n10+n11+n12+n13+n14+n15-
15);
W=(n1+n2+n3+n4+n5+n6+n7+n8+n9+n10+n11+n12+n13+n14+n15-15)*Sc;
Bo=((MX1-MG)'*(MX1-MG))+((MX2-MG)'*(MX2-MG))+((MX3-MG)'*(MX3-
MG))+((MX4-MG)'*(MX4-MG))+((MX5-MG)'*(MX5-MG))+((MX6-MG)'*(MX6-
MG))+((MX7-MG)'*(MX7-MG))+((MX8-MG)'*(MX8-MG))+((MX9-MG)'*(MX9-
MG))+((MX10-MG)'*(MX10-MG))+((MX11-MG)'*(MX11-MG))+((MX12-
MG)'*(MX12-MG))+((MX13-MG)'*(MX13-MG))+((MX14-MG)'*(MX14-
MG))+((MX15-MG)'*(MX15-MG)) ;
end
wb=inv(W)*Bo;
nwb=length(diag(wb));
[a,B1]=eig(wb);
B1=diag(B1);
[B1,il]=sort(B1);
lambda=diag(flipud(B1));
il=flipud(il);
il=il';
a=a(:,il');
for i=1:nwb
    al=a(:,i);
    cl=al'*Sc*al;
    k=1/sqrt(cl);
    e(i,:)=(-1*k*al)';
end
MY=MX*e';
YE=x*e';
for i=1:m
    for j=1:N
        d1(j,:)=YE(i,:)-MY(j,:);
        for k=1:nwb
            d(j,k)=(d1(j,k))^2;
            sd(j)=sqrt(sum(d(j,:)));
        end
        md1=min(sd);
        sd1(i,j)=sd(j);
    end
    md(i)=md1;
end
for i=1:m
    for j=1:N
        if md(i)==sd1(i,j)
            yf(i,:)=j;
        end
    end
end

```

```

YF(i,:)=MX(j,:);
    else
    end
end
end
t2f=clock;
dtf=etime(t2f,t1f);
AM=[YF(:,1) YF(:,2) YF(:,3)];
A1=(AM(:,1))';
A2=(AM(:,2))';
A3=(AM(:,3))';
[h1,h2]=size(R);
for j=1:h2
    k1=(j-1)*h1;
    for i=1:h1
        k2=k1+i;
        RRR(i,j)=A1(1,k2);
        GGG(i,j)=A2(1,k2);
        BBB(i,j)=A3(1,k2);
    end
end
end

```

Subrotinas das Imagens à Serem Discriminadas.

```

ALGORITMO:"corr2.m"
X=[1 1 1 1 1 1 1 2 2 2 2 2 2 2 2
    1 1 1 1 1 1 1 2 2 2 2 2 2 2 2
    1 1 1 1 1 1 1 2 2 2 2 2 2 2 2
    1 1 1 1 1 1 1 2 2 2 2 2 2 2 2
    1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
xx=(X(:))';
map=[(Y1(:))' ; (Y2(:))'];
[R,G,B]=ind2rgb(X,map);
x=[R(:) G(:) B(:)];
Y=[Y1;Y2];
ALGORITMO: "corr3.m"
X=[1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
    1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
    1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
    1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
    1 1 1 1 1 2 2 2 2 2 3 3 3 3 3];
xx=(X(:))';
map=[(Y1(:))' ; (Y2(:))' ; (Y3(:))'];

```

```

[R,G,B]=ind2rgb(X,map);
x=[R(:) G(:) B(:)];
Y=[Y1;Y2;Y3];
ALGORITMO:"corr5.m"
XX=[1 1 1 2 2 2 3 3 3 4 4 4 5 5 5
    1 1 1 2 2 2 3 3 3 4 4 4 5 5 5
    1 1 1 2 2 2 3 3 3 4 4 4 5 5 5
    1 1 1 2 2 2 3 3 3 4 4 4 5 5 5
    1 1 1 2 2 2 3 3 3 4 4 4 5 5 5];
xx=XX(:);
map=[(Y1(:))' ;(Y2(:))' ;(Y3(:))' ;(Y4(:))' ;(Y5(:))'];
[R,G,B]=ind2rgb(XX,map);
x=[R(:) G(:) B(:)];
Y=[Y1;Y2;Y3;Y4;Y5];
ALGORITMO:"corr6.m"
XX=[1 1 2 2 3 3 4 4 4 5 5 5 6 6 6
    1 1 2 2 3 3 4 4 4 5 5 5 6 6 6
    1 1 2 2 3 3 4 4 4 5 5 5 6 6 6
    1 1 2 2 3 3 4 4 4 5 5 5 6 6 6
    1 1 2 2 3 3 4 4 4 5 5 5 6 6 6];
xx=XX(:);
map=[(Y1(:))' ;(Y2(:))' ;(Y3(:))' ;(Y4(:))' ;(Y5(:))' ;(Y6(:))'];
[R,G,B]=ind2rgb(XX,map);
x=[R(:) G(:) B(:)];
Y=[Y1;Y2;Y3;Y4;Y5;Y6];
ALGORITMO:"corr10.m"
XX=[1 2 2 3 4 4 5 6 6 7 8 8 9 10 10
    1 2 2 3 4 4 5 6 6 7 8 8 9 10 10
    1 2 2 3 4 4 5 6 6 7 8 8 9 10 10
    1 2 2 3 4 4 5 6 6 7 8 8 9 10 10
    1 2 2 3 4 4 5 6 6 7 8 8 9 10 10];
xx=XX(:);
map=[(Y1(:))' ;(Y2(:))' ;(Y3(:))' ;(Y4(:))' ;
      (Y5(:))' ;(Y6(:))' ;(Y7(:))' ;(Y8(:))' ;
      (Y9(:))' ;(Y10(:))'];
[R,G,B]=ind2rgb(XX,map);
x=[R(:) G(:) B(:)];
Y=[Y1;Y2;Y3;Y4;Y5;Y6;Y7;Y8;Y9;Y10];
ALGORITMO:"corr15.m"
XX=[ 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
    1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
    1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
    1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
    1 2 3 4 5 6 7 8 9 10 11 12 13 14 15];
xx=XX(:);

```


Subrotinas da Entrada da Imagem do Macaco Mandril.

```

load mandrill
Bi=imcrop(X,[195,200,100,100]);
[R,G,B]=ind2rgb(Bi,map);
imshow(R,G,B)
pause
XX=Bi(:);
[c1,l1]=size(XX)
for i=1:c1
    if XX(i)<50
        xx(i)=2;
    elseif XX(i)>145
        xx(i)=3;
    else
        xx(i)=1;
    end
end
x=[R(:) G(:) B(:)];
Y=[Y1;Y2;Y3];

```

Subrotinas das Amostras de Cada Cor.

```

ALGORITMO:"cor1.m"
c1=[1 1 1 1 1 1 1 1 1 1
    2 2 2 2 2 2 2 2 2 2
    3 3 3 3 3 3 3 3 3 3
    4 4 4 4 4 4 4 4 4 4];
map1;
[r, g, b]=ind2rgb(c1,map1);
X=[r(:) g(:) b(:)];
Y=mean(map1);
ALGORITMO:"cor2.m"
c2=[1 1 1 1 1 1 1 1 1 1
    2 2 2 2 2 2 2 2 2 2
    3 3 3 3 3 3 3 3 3 3
    4 4 4 4 4 4 4 4 4 4];
map2;
[r, g, b]=ind2rgb(c2,map2);
X=[r(:) g(:) b(:)];
Y=[0 128/255 128/255];
ALGORITMO:"cor3.m"
c3=[1 1 1 1 1 1 1 1 1 1
    2 2 2 2 2 2 2 2 2 2

```

```

3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map3;
[r, g, b]=ind2rgb(c3,map3);
X=[r(:) g(:) b(:)];
Y=[128/255 128/255 64/255];
ALGORITMO:"cor4.m"
c4=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map4;
[r, g, b]=ind2rgb(c4,map4);
X=[r(:) g(:) b(:)];
Y=[0 255/255 128/255]
ALGORITMO:"cor5.m"
c5=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map5;
[r, g, b]=ind2rgb(c5,map5);
X=[r(:) g(:) b(:)];
Y=[0 255/255 0];
ALGORITMO:"cor6.m"
c6=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map6;
[r, g, b]=ind2rgb(c6,map6);
X=[r(:) g(:) b(:)];
Y=[255/255 255/255 0];
ALGORITMO:"cor7.m"
c7=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map7;
[r, g, b]=ind2rgb(c7,map7);
X=[r(:) g(:) b(:)];
Y=[0 255/255 255/255];
ALGORITMO:"cor8.m"
c8=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2

```

```

3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map8;
[r, g, b]=ind2rgb(c8,map8);
X=[r(:) g(:) b(:)];
Y=[0 0 255/255];
ALGORITMO:"cor9.m"
c9=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map9;
[r, g, b]=ind2rgb(c9,map9);
X=[r(:) g(:) b(:)];
Y=[255/255 0 255/255];
ALGORITMO:"cor10.m"
c10=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map10;
[r, g, b]=ind2rgb(c10,map10);
X=[r(:) g(:) b(:)];
Y=[1 1 1];
ALGORITMO:"cor11.m"
c11=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map11;
[r, g, b]=ind2rgb(c11,map11);
X=[r(:) g(:) b(:)];
Y=[0 0 0];
ALGORITMO:"cor12.m"
c12=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4];
map12;
[r, g, b]=ind2rgb(c12,map12);
X=[r(:) g(:) b(:)];
Y=[128/255 128/255 128/255];
ALGORITMO:"cor13.m"
c13=[1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2

```

```

3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4];
map13;
[r, g, b]=ind2rgb(c13,map13);
X=[r(:) g(:) b(:)];
Y=[255/255 0 0];
ALGORITMO:"cor14.m"
c14=[1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4];
map14;
[r, g, b]=ind2rgb(c14,map14);
X=[r(:) g(:) b(:)];
Y=[192/255 192/255 192/255];
ALGORITMO:"cor15.m"
c15=[1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4];
map15;
[r, g, b]=ind2rgb(c15,map15);
X=[r(:) g(:) b(:)];
Y=[128/255 255/255 255/255];

```

Subrotinas dos Mapas de Amostras de Cores.

```

ALGORITMO:"mapa1.m"

map1=[ 0 128 0; 1 129 1; 2 130 2; 0 128 0]/255;
map2=[ 0 128 128; 1 129 129; 2 130 130; 0 128 128]/255;
map3=[128 128 64;129 129 65;130 130 66;128 128 64]/255;
map4=[ 0 255 128; 1 254 129; 2 253 130; 0 255 128]/255;
map5=[ 0 255 0; 1 254 1; 2 253 2; 0 255 0]/255;
map6=[255 255 0;254 254 1;253 253 2;255 255 0]/255;
map7=[ 0 255 255; 1 254 254; 2 253 253; 0 255 255]/255;
map8=[ 0 0 255; 1 1 254; 2 2 253; 0 0 255]/255;
map9=[255 0 255;254 1 254;253 2 253;255 0 255]/255;
map10=[255 255 255;254 254 254;253 253 253;255 255 255]/255;
map11=[ 0 0 0; 1 1 1; 2 2 2; 0 0 0]/255;
map12=[128 128 128;129 129 129;130 130 130;128 128 128]/255;
map13=[255 0 0;254 1 1;253 2 2;255 0 0]/255;
map14=[192 192 192;193 193 193;194 194 194;192 192 192]/255;
map15=[128 255 255;129 254 254;130 253 253;128 255 255]/255;

```

ALGORÍTMO:"mapa2.m"

```

map1=[ 12 140 12; 10 138 10; 11 139 11; 12 140 12]/255;
map2=[ 12 140 140; 10 138 138; 11 139 139; 12 140 140]/255;
map3=[140 140 82;138 138 74;139 139 75;130 130 82]/255;
map4=[ 12 247 140; 10 245 138; 11 246 139; 12 247 140]/255;
map5=[ 12 247 12; 10 245 10; 11 246 11; 12 247 12]/255;
map6=[247 247 12;245 245 10;246 246 11;247 247 12]/255;
map7=[ 12 247 247; 10 245 245; 11 246 246; 12 247 247]/255;
map8=[ 12 12 247; 10 10 245; 11 11 246; 12 12 247]/255;
map9=[247 12 247;245 10 245;246 11 246;247 12 247]/255;
map10=[247 247 247;245 245 245;246 246 246;247 247 247]/255;
map11=[ 12 12 12; 10 10 10; 11 11 11; 12 12 12]/255;
map12=[140 140 140;138 138 138;139 139 139;140 140 140]/255;
map13=[247 12 12;245 10 10;246 11 11;247 12 12]/255;
map14=[204 204 204;202 202 202;203 203 203;204 204 204]/255;
map15=[140 247 247;138 245 245;139 246 246;140 247 247]/255;

```

ALGORÍTMO:"mapa3.m"

```

map1=[ 0 128 0; 1 129 1; 11 139 11; 12 140 12]/255;
map2=[ 0 128 128; 1 129 129; 11 139 139; 12 140 140]/255;
map3=[128 128 64;129 129 65;139 139 75;140 140 82]/255;
map4=[ 0 255 128; 1 254 129; 10 245 138; 12 247 140]/255;
map5=[ 0 255 0; 1 254 1; 10 245 10; 12 247 12]/255;
map6=[255 255 0;254 254 1;245 245 10;247 247 12]/255;
map7=[ 0 255 255; 1 254 254; 10 245 245; 12 247 247]/255;
map8=[ 0 0 255; 1 1 254; 10 10 245; 12 12 247]/255;
map9=[255 0 255;254 1 254;245 10 245;247 12 247]/255;
map10=[255 255 255;254 254 254;245 245 245;247 247 247]/255;
map11=[ 0 0 0; 1 1 1; 10 10 10; 12 12 12]/255;
map12=[128 128 128;129 129 129;138 138 138;140 140 140]/255;
map13=[255 0 0;254 1 1;245 10 10;247 12 12]/255;
map14=[192 192 192;193 193 193;202 202 202;204 204 204]/255;
map15=[128 128 128;129 254 254;138 245 245;140 247 247]/255;

```

ALGORÍTMO:"mapa4.m"

```

map1a=[ 12 140 12; 10 138 10; 11 139 11; 12 140 12]/255;
map1b=[ 0 128 0; 1 129 1; 2 130 2; 0 128 0]/255;
map1=map1a;
map2a=[ 12 140 140; 10 138 138; 11 139 139; 12 140 140]/255;
map2b=[ 0 128 128; 1 129 129; 2 130 130; 0 128 128]/255;
map2=map2b;
map3a=[140 140 82;138 138 74;139 139 75;140 140 82]/255;
map3b=[128 128 64;129 129 65;130 130 66;128 128 64]/255;
map3=map3a;
map4a=[ 12 247 140; 10 245 138; 11 246 139; 12 247 140]/255;
map4b=[ 0 255 128; 1 254 129; 2 253 130; 0 255 128]/255;
map4=map4a;
map5a=[ 12 247 12; 10 245 10; 11 246 11; 12 247 12]/255;
map5b=[ 0 255 0; 1 254 1; 2 253 2; 0 255 0]/255;
map5=map5a;
map6a=[247 247 12;245 245 10;246 246 11;247 247 12]/255;
map6b=[255 255 0;254 254 1;253 253 2;255 255 0]/255;
map6=map6a;
map7a=[ 12 247 247; 10 245 245; 11 246 246; 12 247 247]/255;
map7b=[ 1 255 255; 1 254 254; 2 253 253; 1 255 255]/255;
map7=map7a;
map8a=[ 12 12 247; 10 10 245; 11 11 246; 12 12 247]/255;
map8b=[ 0 0 255; 1 1 254; 2 2 253; 0 0 255]/255;
map8=map8a;
map9a=[247 12 247;245 10 245;246 11 246;247 12 247]/255;
map9b=[255 0 255;254 1 254;253 2 253;255 0 255]/255;
map9=map9a;
map10a=[247 247 247;245 245 245;246 246 246;247 247 247]/255;
map10b=[255 255 255;254 254 254;253 253 253;255 255 255]/255;
map10=map10b;
map11a=[ 12 12 12; 10 10 10; 11 11 11; 12 12 12]/255;
map11b=[ 0 0 0; 1 1 1; 2 2 2; 0 0 0]/255;
map11=map11b;
map12a=[140 140 140;138 138 138;139 139 139;140 140 140]/255;
map12b=[128 128 128;129 129 129;130 130 130;128 128 128]/255;
map12=map12b;
map13a=[247 12 12;245 10 10; 246 11 11;247 12 12]/255;
map13b=[255 0 0;254 1 1;253 2 2;255 0 0]/255;
map13=map13b;
map14a=[204 204 204;202 202 202;203 203 203;204 204 204]/255;
map14b=[192 192 192;193 193 193;194 194 194;192 192 192]/255;
map14=map14b;

```

```

map15a=[140 247 247;138 245 245;139 246 246;140 247 247]/255;
map15b=[128 255 255;129 254 254;130 253 253;128 255 255]/255;
map15=map15b;

```

ALGORÍTMO:"mapa5.m"

```

map1a=[ 0 128 0; 1 129 1; 11 139 11; 12 140 12]/255;
map1b=[ 0 128 0; 1 129 1; 2 130 2; 0 128 0]/255;
map1=map1b;
map2a=[ 0 128 128; 1 129 129; 11 139 139; 12 140 140]/255;
map2b=[ 0 128 128; 1 129 129; 2 130 130; 0 128 128]/255;
map2=map2b;
map3a=[128 128 64;129 129 65;139 139 75;140 140 82]/255;
map3b=[128 128 64;129 129 65;130 130 66;128 128 64]/255;
map3=map3b;
map4a=[ 0 255 128; 1 254 129; 10 245 138; 12 247 140]/255;
map4b=[ 0 255 128; 1 254 129; 2 253 130; 0 255 128]/255;
map4=map4b;
map5a=[ 0 255 0; 1 254 1; 10 245 10; 12 247 12]/255;
map5b=[ 0 255 0; 1 254 1; 2 253 2; 0 255 0]/255;
map5=map5b;
map6a=[255 255 0;254 254 1;245 245 10;247 247 12]/255;
map6b=[255 255 0;254 254 1;253 253 2;255 255 0]/255;
map6=map6b;
map7a=[ 0 255 255; 1 254 254; 10 245 245; 12 247 247]/255;
map7b=[ 1 255 255; 1 254 254; 2 253 253; 1 255 255]/255;
map7=map7b;
map8a=[ 0 0 255; 1 1 254; 10 10 245; 12 12 247]/255;
map8b=[ 0 0 255; 1 1 254; 2 2 253; 0 0 255]/255;
map8=map8b;
map9a=[255 0 255;254 1 254;245 10 245;247 12 247]/255;
map9b=[255 0 255;254 1 254;253 2 253;255 0 255]/255;
map9=map9b;
map10a=[255 255 255;254 254 254;245 245 245;247 247 247]/255;
map10b=[255 255 255;254 254 254;253 253 253;255 255 255]/255;
map10=map10b;
map11a=[ 0 0 0; 1 1 1; 10 10 10; 12 12 12]/255;
map11b=[ 0 0 0; 1 1 1; 2 2 2; 0 0 0]/255;
map11=map11b;
map12a=[128 128 128;129 129 129;138 138 138;140 140 140]/255;
map12b=[128 128 128;129 129 129;130 130 130;128 128 128]/255;
map12=map12b;
map13a=[255 0 0;254 1 1;245 10 10;247 12 12]/255;
map13b=[255 0 0;254 1 1;253 2 2;255 0 0]/255;

```

```
map13=map13b;  
map14a=[192 192 192;193 193 193;202 202 202;204 204 204]/255;  
map14b=[192 192 192;193 193 193;194 194 194;192 192 192]/255;  
map14=map14b;  
map15a=[128 128 128;129 254 254;138 245 245;140 247 247]/255;  
map15b=[128 255 255;129 254 254;130 253 253;128 255 255]/255;  
map15=map15a;
```