

Francisco do Nascimento Júnior
Marinas Carrasco de Ribamar Dantas
Patrícia Maforte dos Santos

*ImageSelector: um classificador de
imagens*

Orientador:
Alejandro Frery

UIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA

Recife

2002

Lista de Figuras

1	Exemplo de imagem gráfica	p. 7
2	Exemplo de imagem fotográfica	p. 7
3	Exemplo de imagem fotográfica para verificação de características . . .	p. 14
4	Exemplo de imagem gráfica para verificação de características	p. 15
5	Exemplo de imagem gráfica GIF	p. 24
6	Exemplo de imagem fotográfica GIF	p. 25
7	Exemplo de imagem gráfica JPG	p. 25
8	Exemplo de imagem gráfica JPG	p. 25
9	Exemplo de imagem GIF com classificação errada	p. 27
10	Gráficos das métricas para imagens GIF - Tela1	p. 30
11	Gráficos das métricas para imagens GIF - Tela2	p. 31
12	Gráficos das métricas para imagens JPG - Tela1	p. 31
13	Gráficos das métricas para imagens JPG - Tela2	p. 32
14	Exemplo do uso da transparência	p. 34
15	Comparação entre GIF e JPG	p. 36
16	Comparação da qualidade de uma imagem JPG	p. 36

Lista de Tabelas

1	Matriz de confusão	p. 22
2	Base de imagens	p. 23
3	Classificação de imagens GIF	p. 26
4	Classificação de imagens JPG utilizando número de cores	p. 26
5	Classificação de imagens JPG sem número de cores	p. 26
6	Matriz de confusão para imagens GIF	p. 27
7	Matriz de confusão para imagens GIF	p. 27

Sumário

1	Resumo	p. 7
2	Objetivos	p. 9
3	Motivação	p. 10
4	Cronograma	p. 12
5	Histórico	p. 13
5.1	Obtenção da base de dados	p. 13
5.2	Estudo da linguagem R	p. 13
5.3	Estudo das métricas	p. 14
5.3.1	Vizinho Mais Distante	p. 15
5.3.2	Histograma de Vizinho Mais Distante	p. 15
5.3.3	Número de Cores	p. 16
5.3.4	Histograma de Cor	p. 16
5.3.5	Porcentagem da Cor Predominante	p. 17
5.3.6	Saturação	p. 17
5.3.7	Razão de Dimensão	p. 18
5.3.8	Dimensão Menor	p. 18
5.4	Estudo dos algoritmos de clustering	p. 18
5.4.1	Método por particionamento	p. 18
5.4.1.1	Partitioning Around Medoids (PAM)	p. 18

5.4.1.2	Clustering Large Applications (CLARA)	p. 19
5.4.1.3	Fuzzy Analysis (FANNY)	p. 19
5.4.2	Método hierárquico	p. 20
5.4.3	Algoritmo escolhido: Clustering Large Applications	p. 20
5.5	Validação do classificador	p. 21
6	Resultados experimentais	p. 23
6.1	Manipulação das imagens	p. 23
6.2	Cálculo das métricas	p. 23
6.3	Classificação	p. 24
6.4	Validação	p. 26
7	Antecedentes	p. 28
7.1	Webseer: um motor de busca de imagens para a www	p. 28
7.2	Distinguindo imagens gráficas e fotográficas na www	p. 28
8	Conclusão	p. 29
Apêndice A – Gráficos das métricas		p. 30
Apêndice B – GIF versus JPG		p. 33
B.1	Formato GIF (Graphics Interchange Format)	p. 33
B.1.1	Visão Geral	p. 33
B.1.2	Principais Características	p. 33
B.1.2.1	Número de Cores	p. 33
B.1.2.2	Controle de Compressão	p. 34
B.1.3	Qualidades	p. 34
B.1.3.1	Transparência	p. 34
B.2	Formato JPEG (Joint Photographic Experts Group)	p. 35

B.2.1	Visão Geral	p. 35
B.2.2	Principais Características	p. 35
	B.2.2.1 Número de Cores	p. 35
	B.2.2.2 Controle de Compressão	p. 35
B.2.3	Qualidades	p. 37
	B.2.3.1 Transparência	p. 37
Apêndice C – GIF Versus JPEG		p. 38
Apêndice D – Formato PPM		p. 39
Referências Bibliográficas		p. 40

1 *Resumo*

O objetivo deste projeto é o desenvolvimento de um classificador que permita classificar imagens coletadas em um banco de dados em duas classes semânticas: imagens fotográficas e imagens gráficas. Imagens fotográficas incluem cenas naturais, com pessoas, faces, animais, flores, paisagens e cidades. Imagens gráficas são logotipos, desenhos, ícones, mapas e panos de fundo, geralmente gerados utilizando o computador.

Na maioria das vezes, não há problema para nós decidirmos se uma imagem é gráfica (Figura 1) ou fotográfica (Figura 2). Baseado nisso, o sistema tentará classificar da mesma maneira que os humanos fariam. No entanto, às vezes não fica claro que categoria pode ser aplicada: é o caso de imagens mistas ou de fotografias de desenhos. Figuras nestes casos não serão utilizadas no treinamento, nem na fase de testes. (ATHITSOS; SWAIN; FRANKE, 1997)

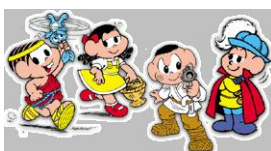


Figura 1: Exemplo de imagem gráfica

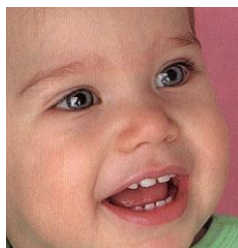


Figura 2: Exemplo de imagem fotográfica

Os classificadores podem ser utilizados em ferramentas comerciais de recuperação de imagens com base no conteúdo.

Meios diferentes podem ser utilizados para coletar as imagens a serem classificadas. Apesar de utilizarmos uma base de dados, uma forma muito interessante e que iremos usar como motivação de projeto é a classificação de imagens da web.

2 *Objetivos*

Serão necessárias as seguintes atividades para o desenvolvimento do classificador:

- leitura da base de imagens do banco de dados;
- estudo da linguagem R, que será utilizada na implementação do classificador;
- estudo, definição, implementação e aplicação de métricas (número de cores, cor predominante, vizinho mais distante, saturação, histograma de cor, histograma do vizinho mais distante, razão de dimensão, dimensão menor) baseadas nas diferenças entre as duas classes em questão;
- disponibilização dos valores resultantes das métricas em uma tabela de atributos com valores normalizados e utilização desta tabela para a classificação das imagens, utilizando o algoritmo Clustering Large Applications. (KAUFMAN; ROUSSEEUW, 1990)
- validação do clasificador, utilizando grupos de 5% de cada categoria para análise de resultados em uma tabela de confusão.

Após o agrupamento das imagens nas categorias desejadas, o sistema permitirá a visualização das métricas para as figuras classificadas através de gráficos.

3 *Motivação*

A Web é uma complexa e vasta fonte de informação multimídia. A cada dia, mais imagens digitais são acrescentadas e retiradas da web, tornando a necessidade de encontrar um determinado grupo de imagens uma tarefa cada vez mais difícil. Desta forma, para se fazer uso dessas informações de forma eficiente é necessário que estas estejam organizadas, de forma a permitir a pesquisa e a recuperação. (FRANKEL; SWAIN; ATHITSOS, 1996)

A recuperação de imagens representa uma área muito ativa desde a década de 1970. Existem duas linhas de pesquisa que estudam a recuperação de imagens, que são o gerenciamento de banco de dados e a visão computacional. Estas duas áreas estudam a recuperação de imagens sob dois ângulos diferentes, um baseado em texto e outro baseado no conteúdo visual das imagens. A recuperação de imagens baseada em textos primeiro descreve as imagens de forma textual e depois utiliza um SGBD (Sistema de Gerenciamento de Banco de Dados) baseado em textos para a recuperação das imagens. Porém, existem algumas dificuldades neste tipo de recuperação. A primeira delas seria a quantidade de trabalho requerida na anotação manual das imagens. A segunda e mais importante seria o aspecto da subjetividade relacionada com a análise do conteúdo das imagens. Ou seja, para uma mesma imagem pessoas diferentes têm percepções diferentes, que ocasionariam anotações diferentes, podendo levar a erros irrecuperáveis no processo de recuperação.

Neste cenário, a organização e recuperação de imagens baseada no conteúdo tornou-se uma área muito importante da visão computacional e da multimídia. A necessidade de se encontrar uma imagem dentro de um conjunto de informações digitais é importante para diversos grupos de usuários, como jornalistas, engenheiros, historiadores, designers, professores, artistas e agências de propagandas, entre outros. A tecnologia que permite o acesso a essas informações vem mudando rapidamente, assim como a forma como as pessoas interagem com a informação visual.

O desenvolvimento de ferramentas que permitam o agrupamento de imagens em gru-

pos semanticamente significativos tornou-se algo de extrema importância, representando uma solução para o desafio da recuperação de imagens baseada no conteúdo. Estas ferramentas tornarão possível a busca eficiente de imagens específicas em um grande banco de dados, pois estas imagens estarão esquematicamente categorizadas.

4 *Cronograma*

1. 11/11/02 – 25/11/02: Obtenção da Base de dados;
Estudo e definição das métricas de classificação;
Estudo do algoritmo de clustering.
2. 26/11/02 – 30/11/02: Estruturação da implementação.
3. 01/12/02 – 20/12/02: Implementação lógica.
4. 21/12/02 – 01/01/03: Recesso Acadêmico.
5. 02/01/03 – 15/01/03: Implementação lógica.
6. 16/01/03 – 08/02/03: Implementação gráfica.
7. 09/02/03 – 11/02/03: Preparação da documentação e apresentação do projeto.

5 *Histórico*

Agora, iremos detalhar melhor como foram atingidas as etapas de desenvolvimento, seguindo uma ordem cronológica:

5.1 **Obtenção da base de dados**

Tentamos entrar em contato com o autor do artigo no qual estamos nos baseando (Oliveira, 2001), mas não obtivemos resposta. Então, procuramos na web uma base de imagens já anteriormente utilizada com a mesma finalidade (classificação), porém também não obtivemos êxito. A solução encontrada foi criar uma base, a partir de CDs de imagens (fotos e clipart's), totalizando 200 imagens, entre as quais 100 são gráficas e 100 são fotográficas. As imagens utilizadas estão em apenas um dos dois formatos, GIF ou JPG.

5.2 **Estudo da linguagem R**

Foi feito um estudo da linguagem e do ambiente R focalizando as funcionalidades oferecidas por esta linguagem para o desenvolvimento do nosso projeto. A linguagem R é bastante eficaz na manipulação de dados, fornecendo técnicas gráficas e estatísticas. O ambiente R fornece facilidades gráficas para a análise e visualização de dados em manipulação. (VENABLES; SMITH; R DEVELOPMENT CORE TEAM, 2002)

Para o desenvolvimento do nosso projeto, inicialmente encontramos a biblioteca "pixmap", que foi utilizada na leitura e manipulação das imagens. Esta biblioteca possui métodos que retornam as matrizes de cores (R, G e B) para a imagem lida, informações que são utilizadas na implementação das métricas. Porém, para que as imagens pudessem ser lidas, foi necessária a conversão das mesmas para o formato .ppm, que representa imagens coloridas da classe de imagens pixmap. Para a conversão das imagens foi utilizado o software ImageMagick, recomendado por um membro da lista de discussão do R. O

ImageMagick é um conjunto de ferramentas e bibliotecas, utilizado para leitura, escrita e manipulação de imagens em diversos formatos.

Na etapa de categorização das imagens iremos utilizar a biblioteca "cluster", que possui funções que implementam os algoritmos de clustering do livro (KAUFMAN; ROUSSEEUW, 1990) para agrupamento de objetos.

5.3 Estudo das métricas

As métricas são parâmetros utilizados para diferenciar os dois tipos de imagens que foram propostos para classificação: imagens gráficas e imagens fotográficas. As principais características apresentadas pelas imagens fotográficas, que podem ser comprovadas através da fig. 3, são:

- Ausência de regiões com cores constantes
- Tendem a apresentar uma pequena diferença na razão altura x largura
- Maior número de cores utilizadas



Figura 3: Exemplo de imagem fotográfica para verificação de características

As principais características apresentadas pelas imagens gráficas, que podem ser comprovadas através da fig. 4, são:

- Grandes regiões com a mesma cor
- Menor número de cores utilizadas
- Tendem a apresentar uma grande diferença na razão largura x altura



Figura 4: Exemplo de imagem gráfica para verificação de características

Sendo assim, as métricas implementadas no projeto foram: vizinho mais distante, razão de dimensão, dimensão menor, saturação, número de cores, porcentagem da cor predominante, histograma de cor e histograma de vizinho mais distante.

5.3.1 Vizinho Mais Distante

A métrica do vizinho mais distante baseia-se na transição de cores que aparece tanto em imagens gráficas como em imagens fotográficas. Imagens gráficas possuem regiões com cores constantes e possuem transições de cores mais visíveis. Desde que o intervalo de cores varie de 0 até 255, o intervalo para a distância de cor varia de 0 até 765.

A distância de cores para dois pontos $p1(r_1, g_1, b_1)$ e $p2(r_2, g_2, b_2)$ é definida como:

$$d = | (r_1 - r_2) | + | (g_1 - g_2) | + | (b_1 - b_2) | . \quad (5.1)$$

Cada pixel da imagem possui vizinhos à esquerda, à direita, acima e abaixo, exceto os pixels das bordas da imagem. O vizinho $p2$ mais distante de $p1$ é aquele em que a distância de cor entre $p2$ e $p1$ maior ou igual do que a distancia entre $p1$ e qualquer um de seus outros vizinhos.

Um parâmetro utilizado nesta métrica é o valor de transição, definido como a distância entre $p1n$ e o seu vizinho mais distante. Outro parâmetro utilizado é o parâmetro P , que pode variar entre 0 e 765. Assim, a métrica do vizinho mais distante para uma imagem é definida como o número de pixels que tem valor de transição maior ou igual a P . (ATHIT-SOS; SWAIN; FRANKE, 1997)

Como as imagens gráficas possuem grandes regiões com a mesma cor, elas possuem muitos pixels cujo valor de transição é zero. Desta forma, se escolhermos o valor de P como sendo igual a 1, é esperado que imagens fotográficas possuam um valor maior do que imagens gráficas para esta métrica.

5.3.2 Histograma de Vizinho Mais Distante

O histograma do vizinho mais distante é um histograma unidimensional com 766 bins. O i -ésimo bin, a partir do zero, contém o número de pixels que possui valor de transição igual à i .

Para calcular esta métrica, cria-se um histograma F_g e F_f , de uma imagem gráfica e de uma imagem fotográfica, respectivamente, pela média dos histogramas do vizinho mais distante de todas as imagens gráficas e fotográficas da base.

O próximo passo é a definição da correlação entre os histogramas A e B como:

$$D(A, B) = \sum_{i=0}^{765} A_i B_i \quad (5.2)$$

onde A_i e B_i são, respectivamente, os i -ésimos bins de A e B .

Para o cálculo do histograma do vizinho mais distante de uma imagem é necessário criar o histograma F_g de uma imagem gráfica, a partir da média dos histogramas do vizinho mais distante de todas as imagens gráficas da base. De maneira similar, é criado o histograma F_f de uma imagem fotográfica a partir do conjunto de imagens fotográficas da base.

O valor calculado s de uma imagem nesta métrica é dado por

$$s = \frac{b}{a + b} \quad (5.3)$$

onde a e b são a correlação entre os histogramas F_i e F_g para a e F_i e F_f para b , com F_i sendo o histograma do do vizinho mais distante para a imagem. Espera-se que imagens fotográficas possuam um valor calculado para esta métrica maior do que o das imagens gráficas.

5.3.3 Número de Cores

O valor utilizado para esta métrica é o número de cores diferentes que aparecem na imagem levando-se em consideração o RGB, ou seja, todos os canais (Red, Green, Blue) juntos. Normalmente, imagens gráficas possuem menor número de cores utilizadas do que imagens fotográficas.

5.3.4 Histograma de Cor

Um histograma de cor é uma tabela tridimensional de tamanho $16 \times 16 \times 16$, no qual cada cor (r, g, b) corresponde ao bin indexado por $(\lfloor \frac{r}{16} \rfloor, \lfloor \frac{g}{16} \rfloor, \lfloor \frac{b}{16} \rfloor)$. Este histograma contém em cada bin o número de pixels na imagem que possuem a cor representada por aquele bin. Para o cálculo desta métrica, é utilizada a correlação $C(A, B)$ entre dois histogramas A e

B, que é definida como

$$C(A, B) = \sum_{i=0}^{16} \sum_{j=0}^{16} \sum_{k=0}^{16} A_{i,j,k} B_{i,j,k}, \quad (5.4)$$

Da mesma forma que na métrica do histograma do vizinho mais distante, para o cálculo do histograma de cores de uma imagem é necessário criar o histograma de cor das imagens gráficas H_g , a partir da média dos histogramas de cores de todas as imagens gráficas da base. De maneira similar, é criado o histograma das imagens fotográficas H_f a partir do conjunto de imagens fotográficas da base. O valor calculado para a imagem I na métrica do histograma de cor é definido como

$$s = \frac{b}{a + b} \quad (5.5)$$

onde a e b são a correlação entre os histogramas H_i e H_g para a e H_i e H_f para b. Espera-se que os valores calculados sejam maiores para imagens fotográficas do que para as imagens gráficas.

5.3.5 Porcentagem da Cor Predominante

O valor calculado para esta métrica é o número de pixels da cor predominante, a cor que aparece em maior quantidade nos pixels da imagem, expresso em porcentagem. Como as imagens gráficas possuem grandes regiões com a mesma cor, o valor calculado para estas imagens é maior do que para imagens fotográficas.

5.3.6 Saturação

O grau de saturação de uma cor é determinado pela quantidade de luz branca nela existente. Quanto maior a quantidade de branco, menor a saturação da cor. Cores altamente saturadas são mais comuns em imagens gráficas do que em imagens fotográficas. Para um pixel p com vetor de cor (r, g, b) seja m o máximo e n o mínimo entre o r , o g e o b . O nível de saturação de p é definido por $|m - n|$. Para o cálculo desta métrica é necessário especificar um parâmetro P , e, o valor calculado é o número de pixels com nível de saturação maior ou igual a P . Para valores altos de P , é esperado que os valores calculados sejam maiores para as imagens gráficas.

5.3.7 Razão de Dimensão

A métrica da razão de dimensão especifica a razão entre a menor dimensão e maior dimensão da imagem. As dimensões devem ser dadas em pixels.

Espera-se que as imagens fotográficas possuam um valor calculado maior que das imagens gráficas.

5.3.8 Dimensão Menor

O valor calculado para uma imagem é o valor de sua menor dimensão em pixels, seja ela largura ou altura. Em geral, as imagens gráficas têm valores calculados menores do que as imagens fotográficas.

Todas as métricas foram normalizadas após o cálculo de seus valores, dividindo-se cada valor obtido por $(largura \times altura)$, que são as dimensões das imagens em pixels.

5.4 Estudo dos algoritmos de clustering

Para escolha do algoritmo de clustering mais adequado ao nosso projeto, foram analisados dois métodos, por particionamento e hierárquico, ambos contidos no livro de Kaufman e Rousseeuw:

5.4.1 Método por particionamento

Neste método, são construídos k grupos, onde cada grupo contém no mínimo um elemento e cada objeto do conjunto pertence a exatamente um grupo. Ele é aplicado quando se deseja classificar objetos em k grupos, onde k é fixo (embora sua escolha possa ser automatizada).

Este método pode ser aplicado através de dois programas: o Partitioning Around Medoids (PAM), o Clustering Large Applications (CLARA) e o Fuzzy Analysis (FANNY).

5.4.1.1 Partitioning Around Medoids (PAM)

Este programa utiliza uma matriz de dissimilaridades entre os objetos do conjunto para identificar os grupos. A idéia do algoritmo é a seguinte:

- selecionam-se k objetos para representar os k grupos (esses objetos levarão a um melhor resultado se forem centralmente localizados);
- cada elemento do conjunto será atribuído ao grupo do objeto representativo mais próximo a ele. Essa medida de proximidade é dada através da distância média ou dissimilaridade média.

Um bom resultado é notado quando a distância média do objeto representativo a todos os demais do seu grupo é mínima, quando isso acontece, chamamos esse objeto ótimo de medóide.

Este algoritmo poderia ser usado para o nosso problema, pois o número de grupos é bem definido. Porém, o limite de objetos a serem categorizados é igual a 100, tornando inviável a construção da matriz de dissimilaridades $n \times n$ caso n ultrapasse esse limite.

5.4.1.2 Clustering Large Applications (CLARA)

Este algoritmo é uma extensão do PAM para conjuntos de entrada grandes, diferenciando apenas no fato de que ele não utiliza a matriz de dissimilaridades entre os objetos. A idéia do CLARA é capturar randomicamente uma amostra do conjunto, buscar os objetos representativos dessa amostra, em seguida, utilizar o PAM para agrupar os objetos. Este algoritmo será utilizado no nosso projeto, pois utilizamos um conjunto razoavelmente grande de objetos.

5.4.1.3 Fuzzy Analysis (FANNY)

Este algoritmo utiliza o princípio da lógica fuzzy, que evita decisões diretas, ou seja, ao invés de afirmar "o objeto A pertence ao grupo 1", ele poderia afirmar "o objeto A pertence por 80% ao conjunto 1 e por 20% ao conjunto 2", significando que provavelmente A pertence ao grupo 1.

Infelizmente a computação deste algoritmo é complexa, não tão óbvia nem intuitiva. Uma outra desvantagem é que ele retorna uma matriz $n \times k$, de n objetos e k grupos, com os coeficientes de pertinência a cada grupo.

5.4.2 Método hierárquico

Esta classe não constrói uma única particionamento com k grupos, mas os objetos se distribuem entre todos os valores de k em uma mesma execução. A princípio, este algoritmo parece ser melhor do que os algoritmos de particionamento, pois ele encontra todas as partições possíveis de uma única vez, restando apenas a escolha de uma delas. Porém devemos levar em consideração que o método de particionamento almeja selecionar o melhor agrupamento de k grupos, o que não é possível com o hierárquico, pois este não pode corrigir uma operação feita anteriormente, impedindo modificações nos agrupamentos. Há dois tipos de métodos hierárquicos: o aglomerativo e o divisivo. No método aglomerativo, partindo de uma iteração de $k = r$ para $k = r + 1$ dois grupos se unem, formando um único grupo. No método divisivo, partindo de uma iteração com $k = r$ para $k = r - 1$, um grupo se divide em dois outros grupos.

Este tipo de algoritmo não se adequa ao nosso problema, pois ele foca problemas que se assemelham com árvores evolucionárias, muito encontrado na biologia, para classificação de plantas, animais, entre outros.

5.4.3 Algoritmo escolhido: Clustering Large Applications

Este algoritmo consiste na seguinte sequência de passos:

1. Selecionam-se cinco ou mais amostras randômicas de objetos de tamanho igual a $40 + 2 * K$, motivado pelo objetivo de se ter uma probabilidade razoável de objetos encontrados para todos os grupos em pelo menos uma amostra.
2. A amostra é agrupada em k categorias usando o algoritmo PAM, que consiste em duas partes:
 - BUILD - Nesta fase são escolhidos sucessivos objetos representativos afim de obter um objeto com a distância média menor possível entre um objeto da amostra e seu objeto representativo mais similar. Isso é feito até se encontrar k objetos representativos.
 - SWAP - Agora, o foco é dado na melhoria do conjunto de objetos representativos e, conseqüentemente, na tentativa de melhorar os agrupamentos gerados por este conjunto. Para isto, é feita a troca de um objeto não-representativo

por um representativo e calculado um fator de agrupamento representando a contribuição daquele objeto no agrupamento caso ele se torne representativo.

3. Após a escolha final dos objetos representativos, cada elemento do conjunto será atribuído a um grupo de acordo com sua proximidade em relação ao objeto representativo do grupo.
4. Calcula-se um fator de qualidade do agrupamento usando a distância média de cada objeto ao representativo do seu grupo.
5. Faz-se os passos 2, 3 e 4 para cada amostra, comparando os seus fatores de qualidade e ressaltando o melhor resultado.

5.5 Validação do classificador

A validação dos dados obtidos na etapa de categorização foi realizada com conjuntos de 5% dos dados de cada conjunto que será classificado. Os resultados obtidos para estes conjuntos serão avaliados através da construção de uma matriz de confusão. Uma matriz de confusão contém informações sobre as classificações reais e previstas feitas por um sistema de classificação. O desempenho de tais sistemas geralmente é avaliado utilizando-se os dados da matriz. A tabela seguinte mostra a matriz de confusão para um classificador de duas classes. As entradas da matriz de confusão têm o seguinte significado no contexto de nosso estudo:

- a é o número de imagens previstas como gráficas e classificadas como gráficas
- b é o número de imagens previstas como gráficas e classificadas como fotográficas
- c é o número de imagens gráficas que foram não foram classificadas
- d é o número de imagens previstas como fotográficas e classificadas como gráficas
- e é o número de imagens previstas como fotográficas e classificadas como fotográficas
- f é o número de imagens fotográficas que foram não foram classificadas

<i>Previstos \ Reais</i>	<i>Gráfica</i>	<i>Fotográfica</i>	<i>Não-clasificadas</i>
<i>Gráfica</i>	a	b	c
<i>Fotográfica</i>	d	e	f

Tabela 1: Matriz de confusão

6 *Resultados experimentais*

6.1 Manipulação das imagens

Inicialmente, buscamos imagens que demonstrassem as mais variadas características de cada classe. Por exemplo, das fotos, buscamos paisagens, pessoas, plantas, objetos; e dentre as gráficas, ícones, cartoons, setas, logotipos, etc. Porém, um outro fator ficou sendo preponderante na escolha das imagens para o estudo: o tamanho. A conversão das imagens para o formato PPM aumenta consideravelmente o tamanho da figura, gerando conseqüências no armazenamento e na leitura das mesmas.

O ambiente R gerou alguns erros na leitura de imagens relativamente grandes (acima de 1Mb no formato PPM), então tornamos público este erro na lista de discussão do R-project e um dos pesquisadores da linguagem nos enviou uma versão corrigida da biblioteca Pixmap que contém as funções de manipulação de imagens.

Em suma, resultamos em 178 imagens, divididas da seguinte forma:

	<i>GIF</i>	<i>JPG</i>	<i>TOTAL</i>
Gráficas	50	39	89
Fotográficas	50	39	89

Tabela 2: Base de imagens

6.2 Cálculo das métricas

Após a separação visual das imagens em dois conjuntos, foram aplicadas as métricas descritas na seção 5.3 a cada um desses conjuntos. Nesta seção, iremos apresentar alguns resultados representativos para cada um desses conjuntos de imagens, diferenciando as imagens gráficas das imagens fotográficas.

Apesar de os valores das métricas terem sido normalizados (como citado na observação

da seção 5.3), eles são apresentados abaixo como obtidos antes da normalização, para que possam ser analisados de maneira mais representativa. Foram aplicadas ao conjunto das imagens GIF todas as métricas descritas na seção 5.3. Porém, ao conjunto das imagens JPG a métrica cor predominante não pôde ser calculada, devido a problemas encontrados com a linguagem de implementação do sistema. Como as imagens JPG possuem uma quantidade de cores muito maior do as imagens GIF, a métrica número de cores só pôde ser calculada através da utilização de alguns artifícios de implementação que impedissem que a quantidade de memória utilizada pela estrutura que armazenava as cores dos pixels já lidos pudesse ultrapassar o limite máximo de memória do tipo de dado desta estrutura. Com isso, não foi possível determinar a cor predominante, pois só foi possível determinar a quantidade de cores diferentes, mas não a frequência total de cada cor. Desta forma, a ordem em que os resultados das métricas são apresentados para as imagens GIF são: vizinho mais distante, razão de dimensão, saturação, número de cores, cor predominante, histograma do vizinho mais distante e histograma de cores. Já para as imagens JPG a ordem é a seguinte: vizinho mais distante, razão de dimensão, saturação, número de cores, histograma do vizinho mais distante e histograma de cores.

É notório que as imagens JPG causaram grande impacto no cronograma de desenvolvimento do sistema, pois a etapa de processamento das métricas é a mais custosa e qualquer erro em seu andamento, prejudicará os passos seguintes. Por causa disso, foi necessário não arriscar quanto a tentativas de buscar soluções mágicas para uma eficiência no cálculo do número de cores (a âncora do atraso), e então formamos um pequeno conjunto de imagens com 21 elementos (11 fotos e 9 gráficas) para vermos o efeito na classifica da ausência das métricas número de cores e cor predominante; o resultado veremos na seção posterior.



Figura 5: Exemplo de imagem gráfica GIF
Métricas = (6542, 26244, 15797, 8, 207072, 17692, 28576, 162)



Figura 6: Exemplo de imagem fotográfica GIF
Métricas = (37229, 39998, 38790, 25, 1759, 23565, 35773, 200)



Figura 7: Exemplo de imagem gráfica JPG
Métricas = (16886, 15375, 14341, 12865, 14485, 21755, 124)



Figura 8: Exemplo de imagem gráfica JPG
Métricas = (43044, 28255, 42412, 31415, 36478, 35724, 168)

6.3 Classificação

Nesta etapa, foi utilizado o algoritmo descrito na seção 5.4.3 para realizar a categorização das imagens nas duas classes desejadas. Não houveram problemas na sua utilização, que, por sinal, mostrou-se ter boa performance.

A saída do método que implementa o algoritmo citado contém algumas informações bem expressivas a respeito do processo realizado pelo algoritmo, como a melhor amostra, os medoides, se os dados foram normalizados, etc.

Abaixo veremos os resultados anotados da classificação em três conjuntos distintos de imagens:

1. Imagens GIF
2. Imagens JPG com número de cores
3. Imagens JPG sem número de cores

<i>Algoritmo \ Real</i>	<i>Gráfica</i>	<i>Fotográfica</i>
Gráfica	44	02
Fotográfica	06	48
Total	50	50

Tabela 3: Classificação de imagens GIF

<i>Algoritmo \ Real</i>	<i>Gráfica</i>	<i>Fotográfica</i>
Gráfica	29	01
Fotográfica	01	26
Total	30	27

Tabela 4: Classificação de imagens JPG utilizando número de cores

<i>Algoritmo \ Real</i>	<i>Gráfica</i>	<i>Fotográfica</i>
Gráfica	09	02
Fotográfica	00	10
Total	09	12

Tabela 5: Classificação de imagens JPG sem número de cores

6.4 Validação

Na etapa de validação, foi observado que a qualidade do classificador, associada ao algoritmo de classificação escolhido, foi bastante satisfatória. A validação da classificação obtida para o conjunto de imagens GIF pode ser observada através da seguinte matriz de confusão:

<i>Previstos \ Reais</i>	<i>Gráfica</i>	<i>Fotográfica</i>	<i>Não-clasificadas</i>
Gráfica	100%	0%	0
Fotográfica	80%	20%	0

Tabela 6: Matriz de confusão para imagens GIF

Os resultados apresentados nesta tabela mostram que a classificação para as imagens gráficas possui um índice de acerto maior, devido à complexidade das imagens gráficas ser menor, levando a valores mais similares para as métricas. Podemos observar na Figura 9 um exemplo de uma imagem fotográfica GIF que foi classificada errada, por apresentar uma pequena quantidade de cores pequena e grandes áreas com a mesma cor.



Figura 9: Exemplo de imagem GIF com classificação errada

Para imagens JPG, os resultados obtidos da validação foram semelhantes. Porém, como o tamanho do conjunto de validação para as imagens JPG é menor em relação ao conjunto das imagens GIF, a matriz de confusão apresenta-se da seguinte forma:

<i>Previstos \ Reais</i>	<i>Gráfica</i>	<i>Fotográfica</i>	<i>Não-clasificadas</i>
Gráfica	100%	0%	0
Fotográfica	80%	20%	0

Tabela 7: Matriz de confusão para imagens GIF

7 *Antecedentes*

7.1 Webseer: um motor de busca de imagens para a www

O Webseer é um sistema para localização de imagens na web. Este sistema utiliza uma indexação das imagens tanto por texto quanto pelo seu conteúdo, pois a análise do conteúdo das imagens acrescenta as informações obtidas através da indexação por texto. Esta informação adicional é utilizada para informar o contexto no qual a imagem está inserida, para facilitar as informações que serão utilizadas pelo algoritmo de classificação.

7.2 Distinguindo imagens gráficas e fotográficas na www

Este sistema foi usado como parte do Webseer, utilizando a análise das imagens pelo seu conteúdo. As imagens a serem classificadas são submetidas a alguns testes, e os resultados são repassados para um classificador.

8 *Conclusão*

Este trabalho contém a definição, desenvolvimento e resultados do problema proposto: a classificação de imagens com base no conteúdo. A justificativa da importância do problema e o motivo de escolha foram comentados já anteriormente no capítulo 3, onde descrevemos a motivação do trabalho. Porém, esta motivação ganha um brilho peculiar após o desenvolvimento e a obtenção dos resultados, pois, foi notável a dificuldade de se trabalhar com imagens e a grande quantidade de dados adquiridos destas.

O classificador obteve bons resultados, categorizando corretamente grande parte das imagens submetidas (ver Tabela de resultados), tendo como principal responsável a atuação do algoritmo de classificação CLARA descrito na seção 5.4.1.2. O que surpreendeu durante a sua execução foi o tempo de processamento para capturar os dados das imagens, porém é simplesmente justificado por se tratar de manipulação de grandes matrizes de dados, além das limitações do ambiente computacional.

Dentre as dificuldades encontradas, pode ser destacada a utilização do ambiente R, não que seja uma linguagem de baixo nível ou limitada, mas pelo fato de ser uma linguagem em desenvolvimento, cujos colaboradores estão sempre atentos às falhas e sugestões indicadas pelos usuários. Com isso, sempre nos deparamos com o conflito de possibilidades do R versus a necessidade na implementação. No entanto, a documentação e a lista de discussão foram pontos importantes para o melhor andamento da implementação.

A longo do trabalho, já foram discutidos os processos, seus resultados e seus problemas, ficando de principal ponto de discussão deste trabalho a variação das escolhas realizadas neste, tais como: métricas, algoritmos de classificação, formatos de figuras, além da indispensável determinação de um ambiente computacional adequado.

APÊNDICE - Gráficos das métricas

Nesta seção iremos mostrar os resultados obtidos a partir das métricas que foram descritas no Sessão 5.3. Os gráficos abaixo irão mostrar a relação entre a quantidade de imagens gráficas e fotográficas cujos valores das métricas foram inferiores a aos dos limiares 0.1, 0.2, ..., até 0.9. Os resultados para cada uma das métricas estão colocados em forma de gráficos, que são apresentados nas Figura 10 a 13. Nas Figuras 10 e 12 são apresentados os resultados para as métricas vizinho mais distante, razão de dimensão, saturação e número de cores. Já nas Figuras 13 e 14, são apresentadas as métricas as métricas restantes, que são cor predominate, histograma do vizinho mais distante e histograma de cores. Estas figuras são telas que foram obtidas como saída do ImageSelector.

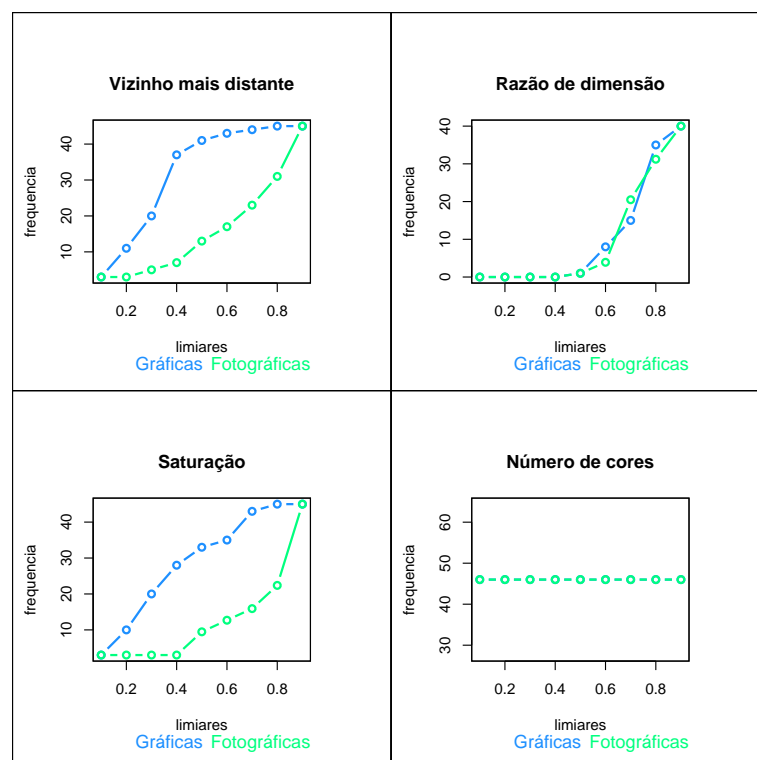


Figura 10: Gráficos das métricas para imagens GIF - Tela1

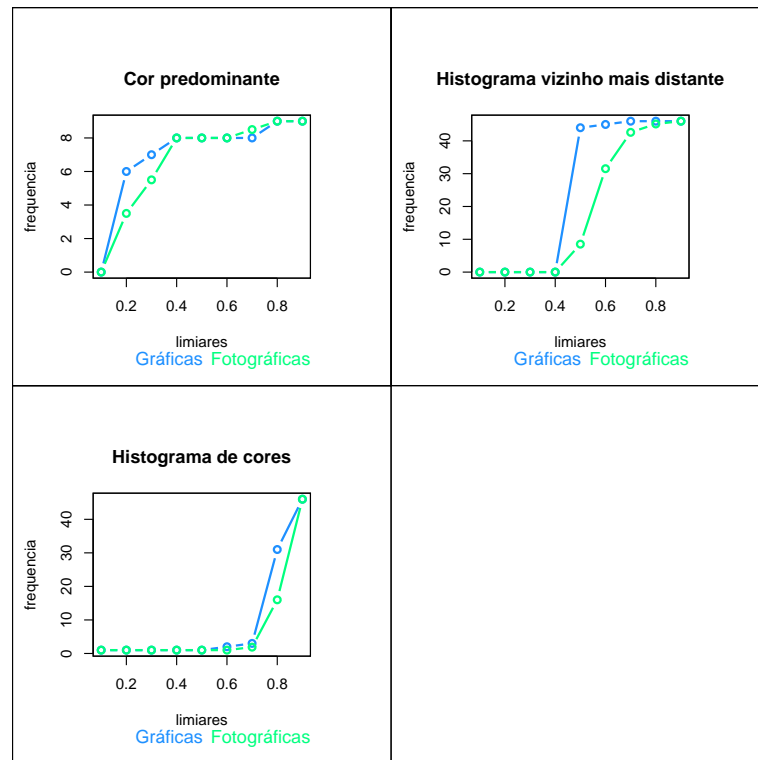


Figura 11: Gráficos das métricas para imagens GIF - Tela2

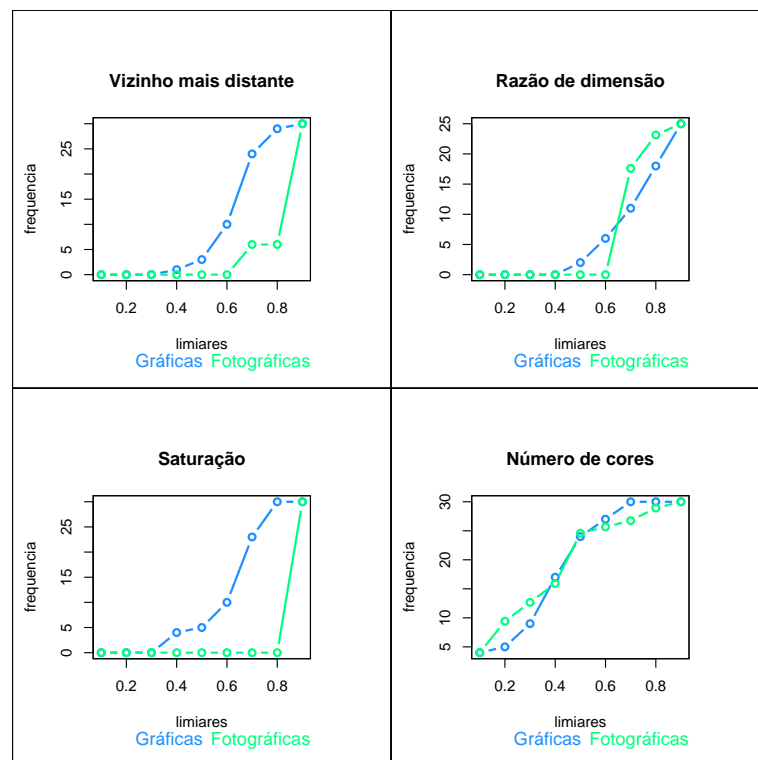


Figura 12: Gráficos das métricas para imagens JPG - Tela1

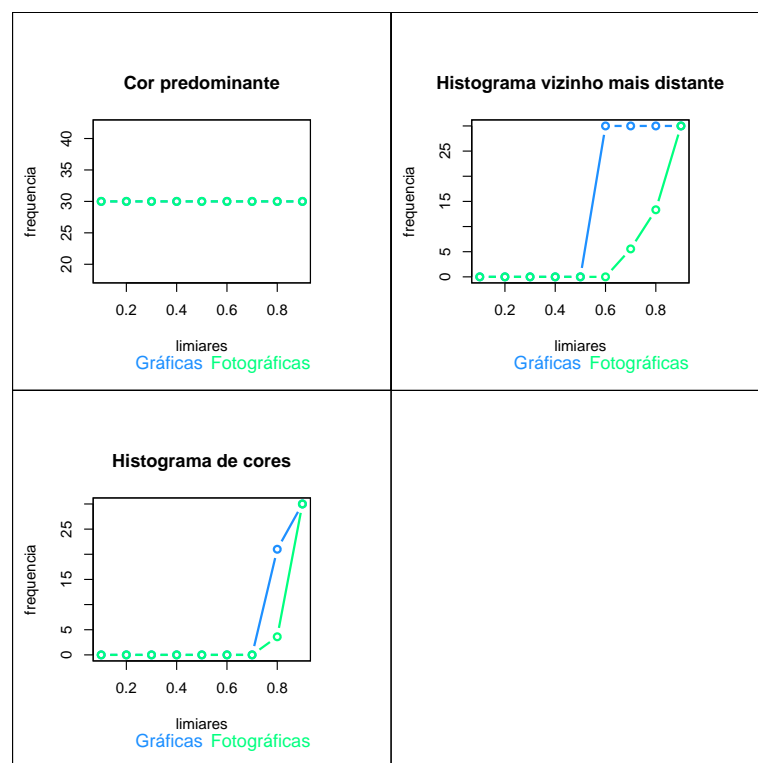


Figura 13: Gráficos das métricas para imagens JPG - Tela2

APÊNDICE A – GIF versus JPG

B.1 Formato GIF (Graphics Interchange Format)

O GIF é diferente dos demais arquivos bitmap comuns pois é baseado em stream. Ele consiste de uma série de pacotes de dados, chamado blocos, que armazenam informação adicional do protocolo. Devido a essa organização, arquivos GIF devem ser lidos como se eles fossem arquivos contínuos de stream de dados.

B.1.1 Visão Geral

Nome: GIF

Também Conhecido Como: Graphics Interchange Format

Tipo: Bitmap

Cores: 1 a 8 bit

Compressão: LZW

Tamanho Máximo Da Imagem: 64Kb X 64Kb pixels

Múltiplas Imagens Por Arquivo: Sim

Formato Numérico: Little-Endian

Originado Por: CompuServe Inc.

Plataforma: MS-DOS, Macintosh, UNIX, Amiga, outros

Aplicações Que Suportam: Muitas

B.1.2 Principais Características

B.1.2.1 Número de Cores

A grande maioria dos arquivos GIF contém imagens com qualidade de 16-cores ou 256-cores. Imagens em tons de cinza, tais como as produzidas por scanners são muitas

vezes armazenadas usando GIF, embora gráficos monocromáticos, tais como clip art e imagens documento, raramente são. Essas cores são indexadas, por isso esse formato funcionará melhor com imagens de cores sólidas, sem gradientes (degradês).

B.1.2.2 Controle de Compressão

O dado imagem armazenado em um arquivo GIF é sempre comprimido no formato LZW. Esse algoritmo reduz strings de valores de bytes idênticos em uma palavra de código único e é capaz de reduzir o tamanho de um dado típico de 8-bit pixel em 40% ou mais. Seu tamanho pode ser reduzido diminuindo-se o número de cores utilizadas para criar uma imagem.

B.1.3 Qualidades

B.1.3.1 Transparência

A forma de uma imagem GIF é sempre retangular, entretanto este formato permite determinar que uma ou mais cores sejam transparentes e portanto não apareçam na tela do browser. Esse recurso transmite a impressão de que a imagem possui, por exemplo, uma forma arredondada. Ao se exportar uma imagem para esse formato tem-se de determinar que a cor de fundo seja transparente.



Figura 14: Exemplo do uso da transparência

Nesse exemplo há três botões: Diving, Charters e Cruises. Os botões Diving e Cruises são GIF's com o fundo transparente e o botão Charters é um JPG que não permite transparência.

B.2 Formato JPEG (Joint Photographic Experts Group)

B.2.1 Visão Geral

Nome: JPEG File Interchange Format

Também Conhecido Como: JFIF, JFI, JPG, JPEG

Tipo: Bitmap

Cores: Acima de 24-bit

Compressão: JPEG

Tamanho Máximo Da Imagem: 64Kx64K pixels

Formato Numérico: Big-endian

Múltiplas Imagens Por Arquivo: Não

Originado Por: C-Clube Microsystems

Plataforma: Todas

Aplicações Que Suportam: Muitas

B.2.2 Principais Características

B.2.2.1 Número de Cores

Arquivos JPG possuem milhões de cores, Truecolor. Por isso funcionarão muito bem para imagens com gradientes (degrades). O JPEG deve ser o formato preferido se você estiver trabalhando com imagens fotográficas que tenham muitas cores tênues e informação tonal. Como você não precisa reduzir as cores de 24 bits para apenas 8 bits de dados (ou 256 cores), você tem uma possibilidade muito maior de reproduções fiéis de imagens com o JPEG. A imagem ilustra uma fotografia salva no formato JPG (acima), com qualidade muito superior ao formato GIF (abaixo) com grande perda de qualidade.

B.2.2.2 Controle de Compressão

A compressão de um arquivo JPG pode ser controlada dosando-se a qualidade da foto versus tamanho do arquivo. Tonar arquivos menores é um grande ganho para transmiti-los entre redes e para o armazenamento de bibliotecas de imagens. Quando se é possível comprimir um arquivo colorido de 2 Mbyte para um de 100 Kbytes ou menos faz grande diferença no espaço em disco e no tempo de transmissão. (Se compararmos GIF e JPEG, a taxa é de quatro para um). No exemplo a mesma imagem é mostrada com duas diferentes



Figura 15: Comparação entre GIF e JPG



Figura 16: Comparação da qualidade de uma imagem JPG

definições de qualidade. A imagem acima está com qualidade máxima (100) e ficaria com 24,43 Kb. A imagem abaixo foi definida para ser exportada com qualidade baixíssima (5) e ficaria com 1,59 Kb.

B.2.3 Qualidades

B.2.3.1 Transparência

Não possui.

APÊNDICE B – GIF Versus JPEG

Mesmo para os aplicativos que suportam JPEG, gasta-se muito mais tempo para decodificar e visualizar uma imagem JPEG do que para uma imagem mais simples como GIF. Assim, usar JPEG é essencialmente um compromisso tempo/espço: você abre mão do tempo para ganhar no armazenamento e transmissão de uma imagem de forma mais barata. Além de ser o melhor formato para fotografias e imagens com nuances de cor, JPG é perfeito para quem precisa dosar a qualidade e o tamanho (em Kb) da imagem.

APÊNDICE C – Formato PPM

O PPM (Portable Pixmap) é um formato simples para imagens a cores. Um ficheiro de imagem em formato PPM-RAWBITS (Portable Pixmap RAWBITS) começa sempre com um cabeçalho:

- a sequência de caracteres P6 e um ou mais caracteres brancos
- a largura l da imagem em pixels e um ou mais caracteres brancos
- a altura h da imagem em pixels e um ou mais caracteres brancos
- o valor máximo de cada componente RGB (que é sempre igual a 255) e um carácter branco
- linhas começadas por $\#$ são comentários (ignorados até ao fim da linha)

Após o cabeçalho seguem-se $3 \times l \times h$ bytes de informação correspondentes aos valores das componentes de cor de cada um dos pixels da imagem (onde l e h são as dimensões da imagem).

Os valores dos pixels são ordenados desde o canto superior esquerdo da imagem até ao canto inferior direito; para cada pixel os valores das componentes são dados pela ordem R,G,B.

Referências Bibliográficas

ATHITSOS, V.; SWAIN, M.; FRANKE, C. Distinguishing photographs and graphics on the World Wide Web. **Workshop in Content-based Access to Image and Video Libraries**, San Juan Puerto Rico, jun. 1997.

FRANKEL, C.; SWAIN, M. J.; ATHITSOS, V. Webseer: An image search engine for the world wide web. **IEEE Computer Vision and Pattern Recognition Conference**, Chicago USA, jun. 1996.

KAUFMAN, L.; ROUSSEEUW, P. J. **Finding groups in data**: An introduction to cluster analysis. New York USA: J. Wiley, 1990.

Oliveira, C. J. S. et al. **Classificação de Imagens Coletadas na Web**. Dissertação (Mestrado) — Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da Computação, Belo Horizonte MG Brasil, 2001.

VENABLES, W. N.; SMITH, D. M.; R DEVELOPMENT CORE TEAM the. **An Introduction to R**. dez. 2002.