

# Classificação de imagens de fotografias históricas usando detecção de linhas e paralelogramos pela presença de edifícios

Natália Cosse Batista

Departamento de Ciência da Computação — Universidade Federal de Minas Gerais  
natalia@dcc.ufmg.br

## Resumo

*Neste trabalho é realizado o estudo de uma metodologia para classificar imagens de fotografias históricas baseada na detecção de edifícios. A metodologia envolve a detecção de bordas pelo algoritmo de Canny e a detecção de linhas e de paralelogramos usando a Transformada de Hough. Devido às características das imagens, são testados também alguns filtros para melhorar a qualidade com redução de ruído e ajuste de contraste. Resultados preliminares mostraram que os atributos usados na classificação possibilitam apenas distinguir uma das classes, a intermediária, das outras duas: edifícios e não-edifícios.*

## 1. Introdução

A preservação de acervos de documentos históricos inclui, muitas vezes, a digitalização desses documentos, de maneira a evitar seu manuseio e conseqüentemente a aceleração do processo de degradação, produzindo centenas de gigabytes de informações. Essas grandes bases de dados necessitam de meios para catalogá-las e organizá-las provendo aos seus usuários - historiadores e pesquisadores, por exemplo - ferramentas para consulta e recuperação dos documentos desejados, de forma fácil e rápida.

As fotografias históricas representam uma parte significativa desse acervo, contendo os mais diversos temas. Apenas no Arquivo Público Mineiro <sup>1</sup> são aproximadamente 80.000 fotografias tipicamente do século XIX e início do século XX, as quais 14.000 encontram-se preparadas para a digitalização e cerca de 6.000 já foram digitalizadas.

---

<sup>1</sup> Com um acervo de aproximadamente 1.400 metros lineares de documentos produzidos e acumulados por órgãos da Administração Pública de Minas Gerais e diversos arquivos privados, o Arquivo Público Mineiro recolhe e conserva importante patrimônio histórico e cultural, abrangendo os séculos XVIII, XIX e parte do século XX.

A anotação manual de um grande acervo é um processo caro e demorado [5], por isso fazem-se necessários sistemas que, além de auxiliarem a anotação, permitam consultas que não estejam restritas a ela. Visando melhorar a indexação e o acesso a esse material, este trabalho propõe a classificação automática das imagens de fotografias pela presença de edifícios, por meio de técnicas de Visão Computacional.

Trabalhos relacionados são discutidos na Seção 2. A metodologia proposta para esse trabalho é descrita na Seção 3. A Seção 4 detalha experimentos preliminares e a Seção 5 traz os resultados iniciais. Algumas conclusões e trabalhos futuros são relatados na Seção 6.

## 2. Revisão Bibliográfica

Os autores de [8] definem edifícios como objetos construídos pelo homem com contornos bem definidos e bordas retas. Essas estruturas contêm, na maioria das vezes, portas e janelas, formando um conjunto de paralelogramos alinhados. Ao passo que linhas extraídas de imagens de não-edifícios são geralmente distribuídas aleatoriamente. A detecção de fotografias que possuem fachadas de edifícios e de outras construções (igrejas, por exemplo), com o auxílio de técnicas de Recuperação de Imagens com Base no Conteúdo Visual (RIBC), pode ser usada na classificação de áreas entre urbanas ou rurais e na diferenciação entre ambientes externos e internos, colaborando com a indexação e recuperação dessas imagens.

As técnicas de RIBC são geralmente baseadas na análise de cores, formas e texturas das imagens. Tipicamente, a construção de um sistema RIBC compreende duas etapas: a representação das imagens e a comparação entre as representações das imagens da base com a imagem-consulta, para computação de um ranking de similaridade [15].

Segundo [9], as técnicas de representação mais usuais podem ser divididas em duas grandes categorias: abordagens baseadas em visão e abordagens baseadas em mode-

los. As abordagens baseadas em visão, aspectos puramente quantitativos da imagem são computados, sendo os mais comuns os histogramas de cores e técnicas de análise de texturas. Por exemplo, as técnicas descritas em [3] e [12], que realizam a classificação de imagens de exteriores. Entretanto, conforme observado por [16], a classificação de imagens que contém estruturas construídas pelo homem não é satisfatória nestes trabalhos pois cor e textura não são informativas para esta classe.

Já as técnicas baseadas em modelos buscam a extração de modelos 3D dos objetos de interesse. Essas técnicas, oriundas principalmente dos trabalhos de visão computacional, requerem um conhecimento *a priori* da forma dos objetos. As técnicas baseadas em modelos são pouco aplicáveis a domínios amplos, pela imprevisibilidade do que pode ser encontrado em cada imagem, não só em termos de conteúdo, mas em termos de escala, ângulo de visão, etc. Em [8], por exemplo, é proposta a aplicação de princípios de agrupamento perceptivo, oriundos da psicologia Gestalt, para extração de elementos de nível intermediário que possam ser vistos como sugestões do conteúdo da imagem. Essa técnica começa pela extração de pequenos segmentos de reta que são, então, agrupados em retas maiores, junções em L, junções em U, grupos paralelos, grupos paralelos significativos e finalmente polígonos. As relações semânticas entre essas estruturas são usadas para detectar a presença de objetos construídos pelo homem, tais como edifícios, torres e pontes.

Em seu trabalho, [6] utilizam visão computacional para compreender cenas que em geral possuem grande diversidade de objetos, como cenas naturais contendo estradas, prédios, árvores, etc. O algoritmo proposto é baseado na detecção de ângulos e retas. Em testes anteriores executados com o programa fornecido pelos autores, nota-se que, em construções, os ângulos de janelas e portas são corretamente marcados, denotando certa organização geométrica, o que não ocorre nas imagens de paisagens naturais e plantações. Aliadas a esses resultados, informações de linhas e paralelogramos contidos nas imagens podem determinar a localização das portas e janelas nas imagens.

O algoritmo de detecção de paralelogramos descrito por [10] é capaz de detectar paralelogramos de tamanhos e orientações variadas mesmo em imagens ruidosas. Nesse trabalho, os autores fazem uso da Transformada de Hough para detectar paralelogramos. Além disso, esse método tem a vantagem de dividir a imagem em partições menores, formando uma grade retangular. Dessa forma, calcular a Transformada de Hough sobre cada partição da imagem reduz a complexidade computacional comparada à Transformada de Hough global. Um algoritmo semelhante é também utilizado por [11] para detectar retângulos.

Com base nessas informações, a metodologia proposta

neste trabalho será composta pela detecção de linhas utilizando a Transformada de Hough e detecção de paralelogramos usando o algoritmo de [10].

### 3. Metodologia proposta

A primeira etapa do trabalho é a seleção de imagens para os experimentos e sua separação em três classes: edifícios, intermediárias e não-edifícios. Essa separação foi sugerida em [8] e contém as seguintes características, adaptadas para esse trabalho:

- Edifícios: maior parte da imagem ocupada por fachada de edifício (parte externa). Pode conter pessoas, árvores, esculturas, carros e outros objetos ocupando pequena parte da imagem.
- Intermediárias: imagem contém alguma parte mínima de fachada, ou edifício distante. Pode conter vários objetos e pessoas ocupando maior porção da imagem.
- Não-edifícios: nenhuma parte da imagem contém edifícios.

O trabalho será realizado utilizando-se um subconjunto do acervo fotográfico digitalizado do APM de diversos temas (construções, paisagens, pessoas, carros, etc) e condições (iluminação, conservação, suporte fotográfico, ângulos, distância da cena, orientação, etc) contendo 90 imagens, 30 de cada classe. As fotografias têm aproximadamente 800x600 pixels em RGB com 24 bits (8 bits para o canal vermelho, 8 bits para o canal verde, 8 bits para azul) para a cor e formato JPEG.

A segunda etapa do trabalho é a detecção de bordas, realizada pelo algoritmo de Canny [1]. A Transformada de Hough será aplicada na imagem das bordas, e usada para a detecção de linhas e paralelogramos<sup>2</sup>. No espaço de Hough, um ponto representa uma linha do plano da imagem e o valor do pico nesse ponto indica o tamanho da linha. O algoritmo de detecção de paralelogramos [10] parte da observação de [7] de que, no espaço de Hough, um paralelogramo é representado por quatro picos que guardam entre si algumas relações geométricas características:

- Esses picos aparecem aos pares, onde cada par tem o mesmo valor de  $\theta$ , ou seja, tem o mesmo ângulo de inclinação;
- As intensidades dos dois picos são as mesmas, pois representam lados opostos do paralelogramo e têm portanto o mesmo tamanho;
- Os pares são separados no eixo  $\theta$  do espaço de Hough por um valor  $\alpha$ , onde  $\alpha$  é um dos ângulos internos do paralelogramo.

---

2 O código-fonte do detector de paralelogramos foi fornecido pelos autores.

- As distâncias verticais entre os picos no eixo  $\rho$  do espaço de Hough são as distâncias entre os lados do paralelogramo.

A terceira etapa do trabalho é a realização de testes preliminares com um conjunto de amostras para determinar parâmetros e limiares. Esse conjunto contém um total de 9 imagens, 3 para cada classe. As métricas escolhidas para os experimentos são:

1. Número total de linhas.
2. Número de linhas maiores ou igual a um limiar.
3. Razão entre as métricas 2 e 1, para efeito de normalização.
4. Número de paralelogramos.

O objetivo inicial é utilizar as métricas 3 e 4 para fazer a separação em classes.

A quarta etapa é a execução do experimento, com os parâmetros e limiares escolhidos na etapa anterior, no conjunto de 90 imagens. Inicialmente, a classificação das imagens será baseada nos limiares escolhidos pela análise dos resultados preliminares. Os resultados serão apresentados por meio de uma matriz de confusão, em que cada linha e cada coluna representam uma classe, cada elemento da matriz contém o número de imagens da classe dada pela linha classificadas como a classe dada pela coluna. A diagonal da matriz de confusão reflete o resultado da classificação para as classes existentes. O esquema de todas as etapas de trabalho é mostrado na Figura 1.

## 4. Experimentos preliminares

### 4.1. Teste inicial

Para um teste inicial, foram selecionadas duas imagens: uma da classe edifícios, outra da classe não-edifícios. Esse primeiro teste foi apenas para ter uma idéia dos valores dos parâmetros. Os parâmetros selecionados para variarem foram:

- tamanho da partição (tile): 10x10, 15x15, 30x30 e  $\text{alt}/2 \times \text{larg}/2$  (alt e larg são a altura e a largura da imagem de entrada, respectivamente).
- lado mínimo do paralelogramo (RMin): 5 e 10.
- ângulo interno do paralelogramo:  $90^\circ$  ou genérico (no algoritmo de [10], existe a possibilidade de não fornecer esse parâmetro, sendo que neste caso procura-se por todos os paralelogramos, independente do ângulo interno).

Como resultado desse teste, observou-se que não foi detectado nenhum paralelogramo de ângulo interno  $90^\circ$  em testes com essas imagens para RMin = 5 e RMin = 10 e



Figura 2. Imagens utilizadas no teste inicial.

tiles 10x10 e 15x15. Nos testes realizados com tiles 10x10, 15x15 e 30x30 e RMin = 5 e RMin = 10 não foram detectados paralelogramos correspondentes às janelas do edifício, mas vários outros que não coincidem com as bordas detectadas. As imagens utilizadas são mostradas na Figura 2 e os resultados para  $\text{tile} = \text{alt}/2 \times \text{larg}/2$  e RMin = 5 estão na Figura 3. Para as imagens testadas, os números de paralelogramos detectados diferem em uma ordem de grandeza.

### 4.2. Experimento com imagens sintéticas

Os resultados da subseção anterior mostraram a necessidade de realizar experimentos com imagens ideais, sintéticas, com ruído mínimo, pois apenas no teste com  $\text{tile} = \text{alt}/2 \times \text{larg}/2$  e RMin = 5 alguns dos paralelogramos detectados estavam de acordo com as bordas oriundas das linhas formadas por portas, janelas e contornos do edifício, sugerindo uma melhor validação do detector para essa função.

Duas imagens sintéticas foram usadas para testar a influência dos parâmetros tamanho da partição (tile) e lado mínimo do paralelogramo (RMin). Os resultados mostraram que tiles próximos do tamanho das janelas da imagem sintética (aproximadamente 16x20 pixels) melhoram a detecção e que o RMin deve ser entre 5 ou 10 para essa imagem. Os testes foram realizados com o detector de paralelogramos configurado para encontrar paralelogramos genéricos. A Figura 4 mostra a tabela com os resultados. Pode-se concluir que os parâmetros que melhor diferenciam as imagens são  $\text{tile} = 20 \times 20$  e RMin = 10. As imagens sintéticas e exemplos de resultados são encontrados na Figura 5.

### 4.3. Experimento com imagens de configurações distintas

Devido às características das imagens, foram testados dois filtros para melhorar a qualidade, o filtro gaussiano para redução de ruído e ajuste de contraste, ambos 3x3. O filtro gaussiano foi usado com  $\sigma = 0,5$  e o filtro de ajuste de contraste com  $\alpha = 0,2$ , e correspondem aos filtros *gaussian* e *unsharp* do programa Matlab (Figura 6).

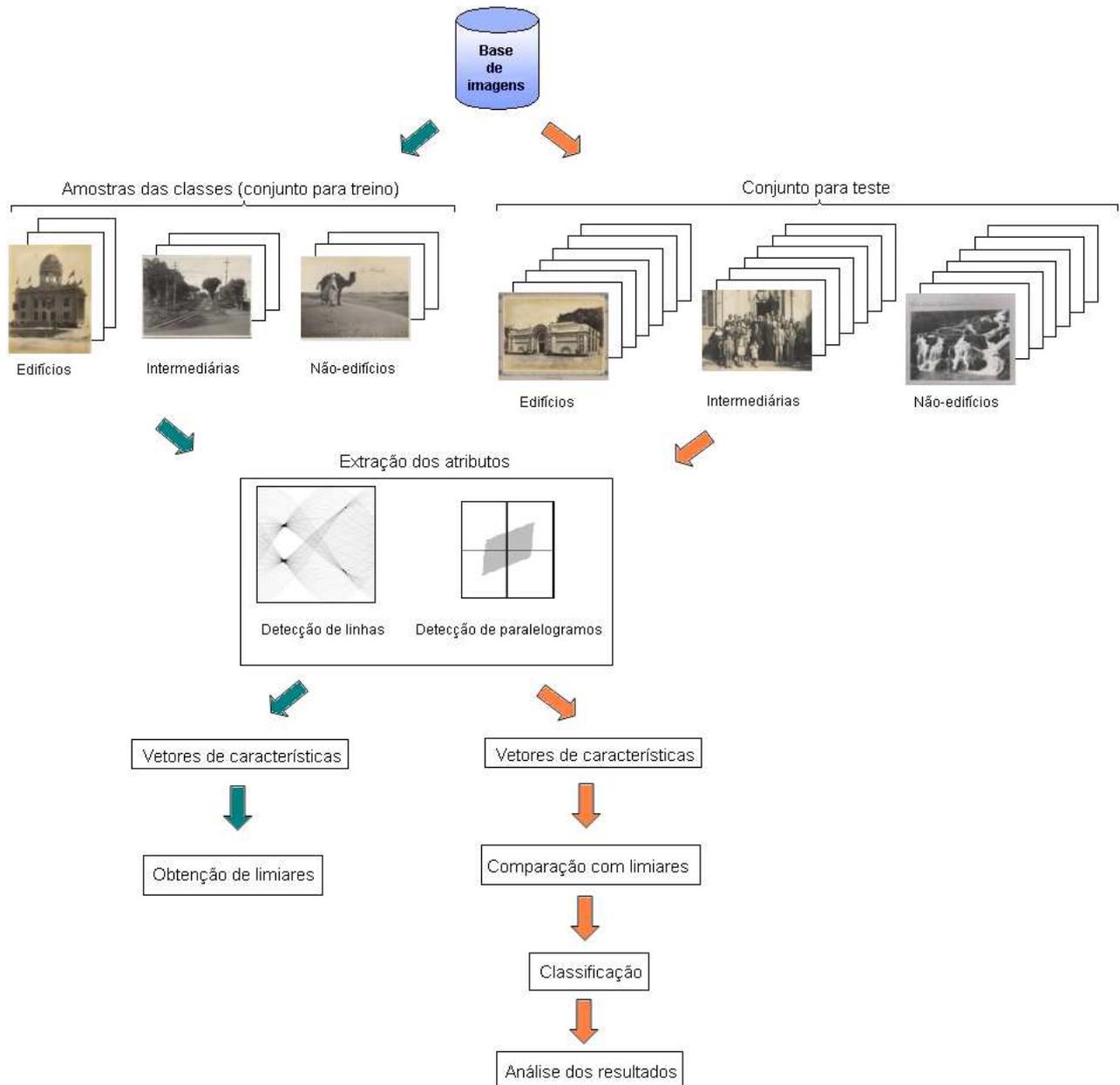


Figura 1. Resumo das etapas do trabalho.

Imagens	# Paralelogramos	# Linhas	# Linhas acima ou igual limiar											
			20	R	40	R	60	R	90	R	100	R		
Edifício	18	84	84	1,00	68	0,80	29	0,34	0	0	0	0	0	
Deserto	2	74	66	0,89	46	0,62	29	0,39	10	0,13	0	0	0	

Figura 3. Resultado do teste inicial. R = métrica 3.

Imagem sintética 1					
Tile	RMin	# linhas	#paralelogramos	# linhas >= 20	pico máximo
5x5	5	1675	5	0	12
10x10	5	586	0	100	20
15x15	5	368	10	113	30
20x20	5	339	33	129	40
20x20	10	245	22	129	40
20x20	15	123	0	112	40
20x20	20	90	0	90	40
30x30	5	177	10	129	60
40x40	5	136	16	115	80
alt/4xlarg/4	5	90	1	82	112
alt/2xlarg/2	5	8	4	8	160
Imagem sintética 2					
Tile	RMin	# linhas	#paralelogramos	# linhas >= 20	pico máximo
5x5	5	2416	2	0	13
10x10	5	1550	20	98	22
15x15	5	1004	71	2002	34
20x20	5	849	36	259	40
20x20	10	276	0	188	40
20x20	15	89	0	86	40
20x20	20	32	0	32	40
30x30	5	576	10	271	65
40x40	5	333	10	192	77
alt/4xlarg/4	5	144	4	109	116
alt/2xlarg/2	5	16	0	14	171

Figura 4. Resultados para as imagens sintéticas.

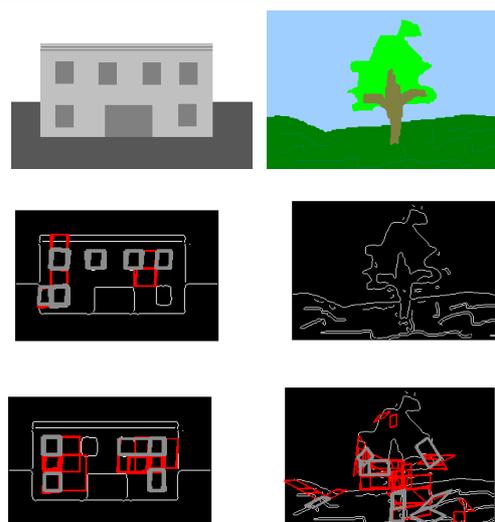


Figura 5. Imagens sintéticas (224 x 145 e 297 x 198, formato JPEG), resultados com tile 20x20 e RMin = 10, resultados com tile 30x30 e RMin = 5. Em cinza os paralelogramos detectados.

gaussian =	0.0113	0.0838	0.0113
	0.0838	0.6193	0.0838
	0.0113	0.0838	0.0113
unsharp =	-0.1667	-0.6667	-0.1667
	-0.6667	4.3333	-0.6667
	-0.1667	-0.6667	-0.1667

Figura 6. Filtros 3x3 gaussiano e ajuste do contraste.

Também foram testadas uma redução de cor para aproximar a fotografia da imagem ideal, com poucos detalhes, uma redução no tamanho das imagens para que o tempo de execução (por volta de 10 minutos por imagem) fosse menor, e a retirada das molduras que introduzem paralelogramos por causa de suas linhas e detalhes, por exemplo na Figura 7. Em [14] é proposta uma metodologia para a remoção automática de molduras da base de imagens do APM, que de fato afetam os resultados da recuperação e indexação de imagens com base no conteúdo.

O objetivo com estes novos testes é verificar o comportamento do detector de linhas e paralelogramos nas imagens com diferentes configurações e visualizar um possível limiar. Nesse experimento foram utilizadas 9 imagens de fotografias, 3 de cada classe. As configurações usadas foram:



**Figura 7. Imagem das bordas com paralelogramos detectados.**

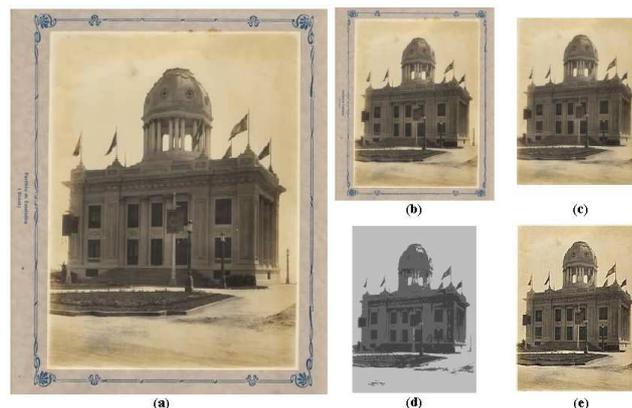
1. Imagem original.
2. Imagem original reduzida em 50% do tamanho.
3. Imagem original reduzida em 50% sem a moldura (retirada manualmente).
4. Imagem original reduzida em 50% sem a moldura com apenas 3 níveis de cinza.
5. Imagem original reduzida em 50% sem a moldura com melhora do contraste e suavização.

O exemplo de uma imagem nas cinco configurações é mostrado na Figura 8.

A Figura 9 mostra os resultados para tile 20x20 e  $R_{Min} = 10$ . Os dados da tabela são o número de imagens resultante da comparação. A configuração 2 foi comparada com a imagem original (configuração 1). A configuração 3 foi comparada com a 2. As outras configurações foram comparadas com os valores da configuração 3. Nota-se que as configurações 2 e 4 reduzem em 100% dos casos as linhas, e em 89% e 78% o número de paralelogramos detectados. A configuração 5 aumenta em 89% das imagens o número de linhas e em 67% os paralelogramos. A configuração 3 em geral reduz o número de linhas detectadas, devido à retirada da moldura.

#### 4.4. Busca de limiares

As mesmas amostras utilizadas na subseção anterior, processadas com tiles 20x20 e  $R_{Min} = 10$ , foram analisadas para a obtenção de um limiar para as métricas 3 e 4. Os resultados estão na Figura 13. Nessa figura, na numeração da primeira coluna, o primeiro algarismo indica a configuração e o segundo, a classe da imagem. Para os limiares 50, 60, 70, 90 e 100 os resultados foram iguais ou próximos de zero e por isso não foram listados. As imagens estão distribuídas da seguinte forma:



**Figura 8. Exemplos das configurações testadas. (a) Imagem original. (b) Imagem original reduzida em 50% do tamanho. (c) Imagem original reduzida em 50% sem a moldura. (d) Imagem original reduzida em 50% sem a moldura com 3 níveis de cinza. (e) Imagem original reduzida em 50% sem a moldura com melhora do contraste e suavização.**

Configuração	# Linhas			# Paralelogramos		
	Aumentou	Reduziu	Manteve	Aumentou	Reduziu	Manteve
2	0	9	0	1	8	0
3	1	7	1	3	4	2
4	0	9	0	2	7	0
5	7	2	0	6	3	0

**Figura 9. Resultados do teste com diferentes configurações das imagens.**

- Edifícios:
  - AB-05-3-012.jpg
  - AB-05-3-015.jpg
  - AB-05-3-033.jpg
- Intermediárias:
  - TG-2-031.jpg
  - TG-1-050.jpg
  - SI-61(30).jpg
- Não-edifícios:
  - AB-06-1-027.jpg
  - JP-11-2-043.jpg
  - AB-08-012(04).jpg

A partir desses resultados, foram estabelecidos limiares para o número de paralelogramos e o número de linhas para duas configurações:

- Imagem original reduzida em 50% sem a moldura
  - Número de paralelogramos acima de 39 e número de linhas abaixo de 689: edifícios.
  - Número de paralelogramos acima de 39 e número de linhas acima de 939: não-edifícios.
  - Número de paralelogramos acima de 39 e número de linhas entre 689 e 939: intermediárias.
  - Número de paralelogramos abaixo de 39 e número de linhas acima de 939: não-edifícios.
  - Número de paralelogramos abaixo de 39 e número de linhas abaixo de 689: intermediárias.
- Imagem original reduzida em 50% sem a moldura com melhora do contraste e suavização
  - Número de paralelogramos abaixo de 55 e número de linhas acima de 971: não-edifícios.
  - Número de paralelogramos abaixo de 55 e número de linhas abaixo de 971: intermediárias.
  - Número de paralelogramos acima de 67 e número de linhas abaixo de 748: edifícios.
  - Número de paralelogramos acima de 67 e número de linhas acima de 748: intermediárias.
  - Número de paralelogramos entre 55 e 67: intermediárias.

## 5. Resultados

Os limiares encontrados na seção anterior foram usados para os testes com as duas configurações escolhidas: imagem original reduzida em 50% sem a moldura (3) e imagem original reduzida em 50% sem a moldura com melhora do contraste e suavização (5). Para o conjunto de teste foram selecionadas aleatoriamente 90 imagens, 30 de cada classe. As imagens sofreram redução de resolução em 50% e as molduras presentes foram retiradas manualmente.

Os resultados são mostrados nas matrizes de confusão das Figuras 10 e 11. Na matriz de confusão da configuração 3, os resultados foram 13%, 57% e 0% para as classes edifícios, intermediárias e não-edifícios, respectivamente, e na configuração 5, os resultados foram 1%, 83% e 0%. Nota-se que os atributos e/ou limiares escolhidos para a classificação possibilitam apenas a distinguir uma das classes, a intermediária, das outras duas: edifícios e não-edifícios.

## 6. Conclusões e trabalhos futuros

Embora o algoritmo de detecção de paralelogramos seja capaz de detectar paralelogramos de tamanhos e orientações variadas mesmo em imagens ruidosas, o número de paralelogramos mostrou-se incapaz de diferenciar as imagens com

Classes	1	2	3	4
1	4	14	6	6
2	3	17	2	8
3	2	24	0	4

**Figura 10. Matriz de confusão para a configuração 3. Classe 1 = edifícios, Classe 2 = intermediárias, Classe 3 = não-edifícios e Classe 4 = não foi possível classificar.**

Classes	1	2	3
1	3	27	0
2	2	25	3
3	0	30	0

**Figura 11. Matriz de confusão para a configuração 5. Classe 1 = edifícios, Classe 2 = intermediárias, Classe 3 = não-edifícios.**

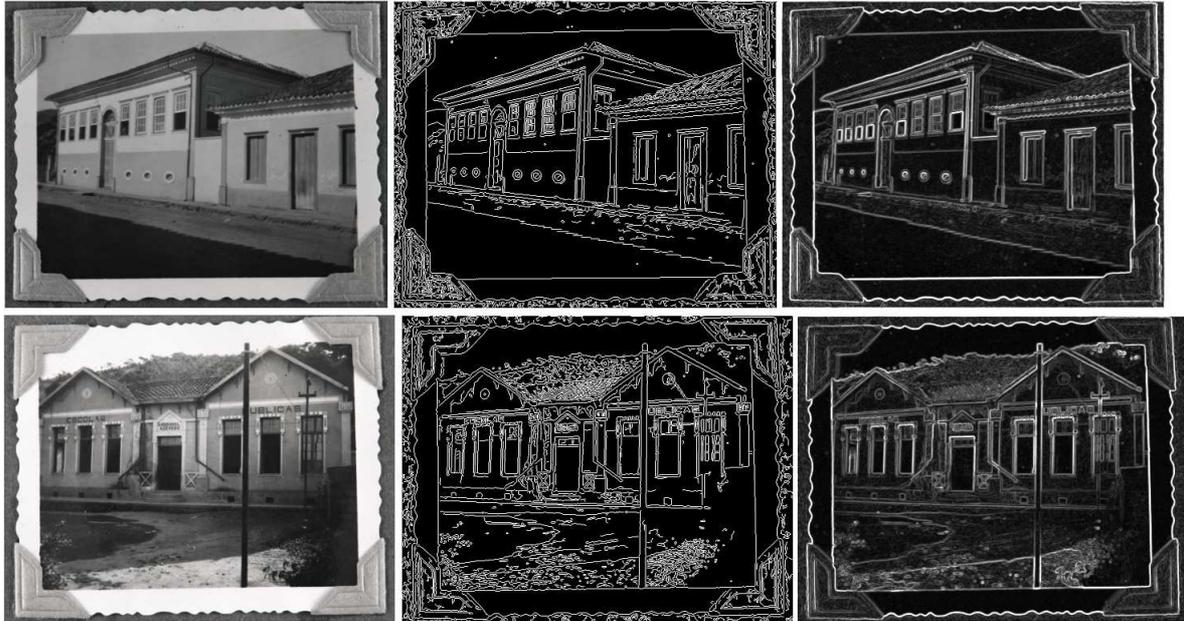
estruturas construídas pelo homem das imagens com estruturas naturais. Um ajuste melhor dos parâmetros  $\text{tile}$  e  $\text{RMIn}$  poderia levar a resultados mais satisfatórios nessas imagens, levando-se em conta uma média do tamanho das janelas, por exemplo.

Aparentemente, os atributos usados na classificação possibilitam distinguir uma das classes, a intermediária, das outras duas, edifícios e não-edifícios. Entretanto, uma análise da matriz de confusão revela que há uma concentração de valores na coluna da classe intermediária, o que significa que a maior parte das imagens foi classificada como intermediária e baixa concentração na diagonal, ou seja, a maioria das imagens não foi classificada corretamente. Isso indica que os atributos escolhidos tem valores próximos para todas as classes.

Como trabalhos futuros, tem-se o objetivo de: utilizar um conjunto de imagens maior para o treinamento e testes, selecionar novos atributos, realizar a classificação com Support Vector Machines (SVM) [2] e fazer a detecção de bordas com o algoritmo de [13]. Esse algoritmo faz a detecção de bordas de Canny com as alterações propostas por [4], e as imagens resultantes são menos ruidosas do que o filtro de Canny implementado no Matlab, como mostra a Figura 12.

## Referências

- [1] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.



**Figura 12. Testes com duas versões do detector de bordas Canny. Imagem original, filtro canny do Matlab, filtro canny alterado.**

- [2] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software disponível em <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [3] X. Feng, C. K. I. Williams, and S. N. Felderhof. Combining belief networks and neural networks for scene segmentation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 24, pages 467–483, 2002.
- [4] M. M. Fleck. Some defects in finite-difference edge finders. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(3):337–345, 1992.
- [5] T. Gevers and A. W. M. Smeulders. *Content-Based Image Retrieval: An Overview*, chapter 8. IMSC Press Multimedia Series. Prentice Hall, 1st edition, July 2004.
- [6] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.
- [7] D. Ioannou and E. Dugan. Parallelogram detection in a digital image with the use of the hough transform. In *Proceedings of the IEEE Computer Society International Conference on Pattern Recognition*, volume II, pages 532–536. IEEE Computer Society, 1996.
- [8] Q. Iqbal and J. Aggarwal. Retrieval by classification of images containing large manmade objects using perceptual grouping. *Pattern Recognition*, 35(7):1463–1479, July 2002.
- [9] Q. Iqbal and J. K. Aggarwal. Feature integration, multi-image queries and relevance feedback in image retrieval. In *Proceedings of 6th International Conference on Visual Information Systems*, pages 467–474, Sep. 24–26 2003.
- [10] C. Jung and S. Rosito. Parallelogram detection using the tiled hough transform. In *Proceedings of 13th International Conference on Systems, Signals and Image Processing*, volume I, pages 177–180, 2006.
- [11] C. R. Jung and R. Schramm. Rectangle detection based on a windowed hough transform. In *Proceedings of XVII Brazilian Symposium on Computer Graphics and Image Processing*, pages 113–120. IEEE Computer Society, 2004.
- [12] S. Konishi and A. L. Yuille. Statistical cues for domain specific image segmentation with performance analysis. In *Proceedings of the 2000 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume I, pages 135–132, 2000.
- [13] P. D. Kovesi. MATLAB and Octave functions for computer vision and image processing. School of Computer Science & Software Engineering, The University of Western Australia. Disponível em <http://www.csse.uwa.edu.au/~pk/research/matlabfns/>.
- [14] A. P. B. Lopes, D. L. Flam, N. C. Batista, S. E. F. de Avila, J. M. Almeiday, and A. de A. Araújo. Frame detection in historical photographs using tiled hough transform. 2008. Submetido para publicação.
- [15] N. Sebe, Q. Tian, E. Louprias, M. Lew, and T. Huang. Evaluation of salient point techniques. *Image and Vision Computing*, 21(13-14):1087–1095, December 2003.
- [16] A. Vailaya, A. K. Jain, and H. J. Zhang. On image classification: City images vs. landscapes. *Pattern Recognition*, 31:1921–1936, 1998.

	Imagens	Paralelo-gramos	# linhas	# linhas >= 20	Razão 20	# linhas >= 30	Razão 30	# linha >= 40	Razão 40	Pico máx
11	AB-05-3-012 .jpg	280	3290	2886	0,877	1480	0,450	474	0,144	48
11	AB-05-3-015 .jpg	176	3060	2723	0,890	1636	0,535	639	0,209	56
11	AB-05-3-033 .jpg	267	3792	3333	0,879	1713	0,452	587	0,155	49
12	SI-61(30) .jpg	87	3910	3056	0,782	1514	0,387	534	0,137	45
12	TG-1-050 .jpg	23	2615	2217	0,848	1132	0,433	443	0,169	47
12	TG-2-031 .jpg	259	3182	2678	0,842	1041	0,327	165	0,052	47
13	AB-06-1-027 .jpg	55	4227	3326	0,787	1115	0,264	288	0,068	48
13	AB-08-012(04) .jpg	259	4188	3356	0,801	911	0,218	40	0,010	44
13	JP-11-2-043 .jpg	119	4442	3648	0,821	1207	0,272	310	0,070	47
21	AB-05-3-012_50 .jpg	115	959	859	0,896	416	0,434	151	0,157	43
21	AB-05-3-015_50 .jpg	98	900	826	0,918	502	0,558	188	0,209	47
21	AB-05-3-033_50 .jpg	76	1021	809	0,792	403	0,395	177	0,173	46
22	SI-61(30)_50 .jpg	18	1332	1096	0,823	552	0,414	178	0,134	60
22	TG-1-050_50 .jpg	25	814	686	0,843	346	0,425	97	0,119	40
22	TG-2-031_50 .jpg	102	862	680	0,789	239	0,277	30	0,035	48
23	AB-06-1-027_50 .jpg	37	1033	877	0,849	332	0,321	90	0,087	41
23	AB-08-012(04)_50 .jpg	42	1343	1166	0,868	474	0,353	56	0,042	50
23	JP-11-2-043_50 .jpg	13	1091	880	0,807	319	0,292	111	0,102	45
31	AB-05-3-012_sm3c50 .jpg	31	548	464	0,847	125	0,228	12	0,022	44
31	AB-05-3-015_sm3c50 .jpg	126	688	598	0,869	300	0,436	53	0,077	59
31	AB-05-3-033_sm3c50 .jpg	65	581	443	0,762	154	0,265	23	0,040	47
32	SI-61(30)_sm3c50 .jpg	5	839	630	0,751	268	0,319	79	0,094	71
32	TG-1-050_sm3c50 .jpg	12	561	464	0,827	174	0,310	14	0,025	44
32	TG-2-031_sm3c50 .jpg	83	723	590	0,816	216	0,299	12	0,017	43
33	AB-06-1-027_sm3c50 .jpg	1	548	445	0,812	150	0,274	23	0,042	42
33	AB-08-012(04)_sm3c50 .jpg	37	1212	1038	0,856	423	0,349	50	0,041	52
33	JP-11-2-043_sm3c50 .jpg	20	817	625	0,765	134	0,164	3	0,004	40
41	AB-05-3-012_sm50 .jpg	154	630	550	0,873	185	0,294	21	0,033	47
41	AB-05-3-015_sm50 .jpg	148	689	600	0,871	302	0,438	58	0,084	60
41	AB-05-3-033_sm50 .jpg	39	653	492	0,753	165	0,253	31	0,047	44
42	SI-61(30)_sm50 .jpg	20	1221	953	0,781	480	0,393	139	0,114	68
42	TG-1-050_sm50 .jpg	25	660	540	0,818	221	0,335	24	0,036	43
42	TG-2-031_sm50 .jpg	66	734	603	0,822	198	0,270	18	0,025	45
43	AB-06-1-027_sm50 .jpg	33	1233	1014	0,822	378	0,307	34	0,028	43
43	AB-08-012(04)_sm50 .jpg	42	1343	1166	0,868	474	0,353	56	0,042	50
43	JP-11-2-043_sm50 .jpg	22	939	722	0,769	201	0,214	5	0,005	40
51	AB-05-3-012_sm50GU .jpg	195	724	675	0,932	291	0,402	50	0,069	56
51	AB-05-3-015_sm50GU .jpg	90	748	670	0,896	385	0,515	92	0,123	60
51	AB-05-3-033_sm50GU .jpg	67	733	591	0,806	256	0,349	55	0,075	60
52	SI-61(30)_sm50GU .jpg	59	1289	1043	0,809	560	0,434	172	0,133	72
52	TG-1-050_sm50GU .jpg	17	658	569	0,865	283	0,430	44	0,067	46
52	TG-2-031_sm50GU .jpg	83	783	683	0,872	304	0,388	57	0,073	55
53	AB-06-1-027_sm50GU .jpg	55	1209	1054	0,872	471	0,390	103	0,085	49
53	AB-08-012(04)_sm50GU .jpg	30	1402	1273	0,908	657	0,469	136	0,097	54
53	JP-11-2-043_sm50GU .jpg	26	971	827	0,852	323	0,333	37	0,038	44

Figura 13. Resultados para as imagens usadas como amostras. As cores (exceto amarelo) indicam possibilidade de determinar um limiar.