

# Detecção e Análise de Conteúdo de Telas de TV e Monitores

Felipe de S. Farias, Luiz E. S. e Silva, Wheidima C. Melo,  
Eddie B. L. Filho, Waldir S. S. Júnior

**Resumo**—Neste artigo, propomos uma metodologia de detecção e extração de telas de TV e monitores que utiliza um modelo baseado no campo aleatório de Markov (MRF). A metodologia proposta envolve diversas etapas, tais como: extração de bordas utilizando gradiente multi-dimensional; detecção de segmentos de linha utilizando a transformada de *Hough* e classificação de segmentos de linha em objetos retangulares. Para os experimentos, utilizamos a metodologia em duas bases de dados (concebidas pelos autores) com, respectivamente, 333 e 230 imagens coloridas de TVs e monitores. Os resultados preliminares demonstram que a metodologia é promissora em ambientes controladas.

**Palavras-Chave**—Detecção de Retângulos, Inspeção Automática, Análise de Imagens.

**Abstract**—In this paper, we propose a detection and extraction method of TV screens and monitors that use a Markov random field (MRF) model. The proposed method involves several steps such as: edge extraction using multi-dimensional gradient; line segments detection using Hough transform and line segments classification (rectangular objects). For the experiments, we apply the methodology in two databases (designed by the authors) with, respectively, 333 and 230 color images of TVs and monitors. Preliminary experiments show that we have promising results.

**Keywords**—Rectangles Detection, Automatic Inspection, Image Analysis.

## I. INTRODUÇÃO

A detecção de objetos retangulares está entre os problemas mais importantes da área de processamento digital de imagens devido ao fato de muitos objetos, feitos por mão humana, terem formato retangular [1]. Entre as técnicas para detecção de retângulos em imagens, podemos citar as que envolvem variações da transformada de *Hough* [2], [3], a análise de informações geométricas de linhas [4], [5] e algoritmos genéticos [6], [7]. Alguns dos problemas enfrentados por estas técnicas são falta de robustez a ruído e o alto custo computacional. As técnicas que utilizam variações da transformada de *Hough* para detecção de retângulos enfrentam o problema do alto custo computacional, além de desempenho fraco com imagens com ruído [1]. As técnicas baseadas em algoritmos genéticos têm como principal problema o alto custo computacional e a dificuldade em encontrar parâmetros que proporcionem bom desempenho de detecção [8].

Felipe de S. Farias, Waldir S. S. Júnior, Eddie B. L. Filho, Pós-graduação em Engenharia Elétrica, Universidade Federal do Amazonas (UFAM) Manaus-AM, Brasil, Eddie B. L. Filho, FPF Tech, Av. Gov. Danilo de Matos Areosa, CEP 69075-351, Manaus-AM, Brasil, Wheidima C. Melo, Escola Superior de Tecnologia, Universidade do Estado do Amazonas (UEA), Manaus-AM, Brasil, Luiz E. S. e Silva, Departamento de Eletricidade, Universidade Federal do Amazonas (UFAM) Manaus-AM, Brasil, E-mails: epiles.farias@gmail.com, luiz.edu.sales@gmail.com, wmelo@uea.edu.br, eddie.filho@fpf.br, waldirjr@ufam.edu.br. Este trabalho foi parcialmente financiado pela CAPES, FAPEAM e Samsung.

A detecção de falhas em TVs e monitores é um problema abordado por diversos trabalhos e alguns destes utilizam técnicas de campos multidisciplinares, como processamento de sinais, engenharia de software, inteligência artificial e reconhecimento de padrões. Por exemplo, no sistema mostrado em [9], os autores propõem encontrar erros no software utilizado no aparelho, tais como *deadlocks*. Outro exemplo é apresentado em [10], onde utiliza-se um sinal de referência misturado com um sinal de erro através de software para encontrar problemas na interação entre hardware e software. Alguns trabalhos de detecção de falhas propõem um sistema de inspeção automática baseado na análise da imagem [11], [12], [13], [14], [15]. Estes sistemas fornecem um diagnóstico da TV (ou monitor) inspecionados através da análise da informação de saída da tela, utilizando esta informação para encontrar falhas de software ou hardware que gerem defeitos na imagem final. Dentre os métodos de inspeção automática de TVs baseados em análise de imagem, destacamos os métodos em que a captura da imagem da TV (ou monitor) é feita através de uma câmera [11], [16]. Devido à variação de tamanho, posição e conteúdo presente na tela dos aparelhos de TV e monitores na ocasião da captura, é necessário um método robusto a tamanho, orientação e cor para detecção de tela e extração do conteúdo da mesma na imagem para o sucesso da inspeção.

Neste artigo, propomos uma metodologia de detecção de telas de TV e monitores e análise do conteúdo extraído que utiliza um modelo baseado no campo aleatório de Markov (MRF). Esta metodologia fornece uma alternativa para o problema de detecção de objetos quadrangulares em imagens coloridas robusta a ruído e a variações na cor e tamanho do objeto. A metodologia proposta envolve diversas etapas, tais como: extração de bordas da imagem através de análise de gradiente multi-dimensional; detecção de segmentos de linha presentes no mapa de bordas da imagem utilizando a transformada de *Hough* e classificação de segmentos de linha que são borda de objetos retangulares usando um algoritmo baseado em um modelo MRF.

### A. Contribuições do Artigo

A principal contribuição deste artigo é a metodologia de detecção e extração de telas em imagens. A metodologia usa um sistema de detecção de telas robusto a variação no tamanho, *background* e cor da tela, técnicas de detecção de borda por análise de gradiente multi-dimensional e detecção de retângulo utilizando um modelo baseado no campo aleatório de Markov. Além desta contribuição, concebemos duas bases de dados com imagens coloridas de TVs e monitores funcionais. A primeira base é composta 333 imagens de TVs e

monitores, capturadas sob condições variáveis de iluminação, posição da câmera e fundo. As imagens são rotuladas com as coordenadas da região que representa a tela. A segunda base é formada por 230 imagens de TVs e monitores, assim como 40 imagens representando o conteúdo apresentado na tela. Neste caso, as imagens foram capturadas em um ambiente com iluminação controlada e posição de câmera e objeto fixos.

## II. METODOLOGIA PROPOSTA

### A. Introdução

A Figura 1 ilustra as etapas do método. A metodologia completa possui quatro etapas, descritas em linhas gerais conforme a seguir: primeiramente, a detecção de bordas é feita através de uma combinação da análise de gradiente multi-dimensional [17] e no detector de Canny [18]. A seguir, no bloco, *detecção de linhas*, um esquema de detecção de linhas baseado na transformada de *Hough* é utilizado no mapa de bordas resultante da etapa anterior. No bloco *detecção de retângulos*, as linhas são agrupadas como parte de objetos retangulares através de um modelo baseado no campo aleatório de Markov (MRF) [1]. Por último, no bloco *escolha de retângulo* os retângulos resultantes são classificados, de forma a encontrar o retângulo que representa a tela. A seguir, detalharemos todas as etapas da metodologia de detecção de telas proposta, enfatizando as suas particularidades, algoritmos e modelagem.

### B. Detecção de Bordas

O esquema de detecção de bordas utilizado neste artigo é baseado na análise de gradiente multi-dimensional [17] e no detector de Canny [18]. Suas etapas, ilustradas na Figura 2, serão descritas conforme segue.

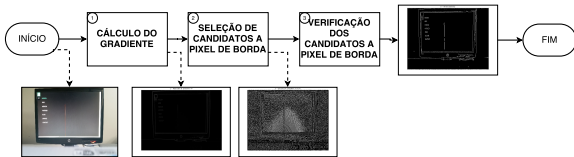


Fig. 2. Diagrama de blocos para detecção de bordas.

Primeiramente, a imagem é tratada como um campo vetorial e são calculadas a magnitude e direção de seu gradiente. A magnitude e a direção são dadas, para cada pixel pertencente à imagem, conforme a seguir:

$$V = \frac{\sqrt{(M_{xx} + M_{yy})^2 - 4(M_{xx}M_{yy} - M_{xy}^2)} + M_{xx} + M_{yy}}{2} \quad (1)$$

$$\theta = \arctan\left(\frac{V - M_{xx}}{M_{xy}}\right) \quad (2)$$

Na equação (1),  $V$  é o valor da magnitude do gradiente em um pixel e  $M_{xx}$ ,  $M_{xy}$  e  $M_{yy}$  são combinações das seis derivadas de primeira ordem da imagem. Na equação (2),  $\theta$  é o valor da direção do gradiente.

Após calcularmos a magnitude e direção de gradiente, um esquema de seleção e verificação de candidatos é implementado para obtermos o mapa de bordas da imagem. No esquema de seleção, um pixel é considerado candidato a borda se

sua magnitude é máxima local, ou seja, se a magnitude de gradiente do pixel é maior que a dos seus vizinhos mais próximos da direção de gradiente.

A verificação é um processo necessário para descartar os candidatos falsos positivos escolhidos na etapa de seleção. A verificação é composta por duas operações que fornecem informações espaciais da imagem regionalmente, conforme descritas a seguir:

- Primeiramente, a vizinhança do pixel é analisada. O candidato a pixel de borda deve ter vizinhos candidatos. Se nenhum dos oito pixels vizinhos for um candidato a pixel de borda, aquele pixel é descartado;
- Não há grande variação entre as magnitudes de pixels de borda vizinhos [1]. Portanto, podemos calcular a diferença entre as magnitudes do gradiente dos pixels vizinhos. Caso esta diferença seja grande, o pixel é descartado. Isto pode ser feito conforme a expressão:

$$\sum_{i,j=0}^{n_{ij}-1} |V(i,j) - V(i+p,j+q)| - \sum_{i,j=0}^{n_{ij}-1} |V(i+p,j+q)| > 0 \quad (3)$$

Na equação (3),  $n_{ij}$  é o número de pixels vizinhos ao candidato a pixel de borda localizado em  $(i, j)$ . Cada vizinho é localizado em  $(i + p, j + q)$ , em que  $p$  e  $q$  pertencem ao intervalo  $\{-1, 0, 1\}$ . Caso a equação (3) seja verdadeira, o candidato a pixel de borda é descartado.

### C. Detecção de Linhas

Neste artigo, montamos um esquema de detecção e combinação de linhas baseado na transformada de *Hough*. As etapas, ilustradas na Figura 3, serão descritas em termos gerais conforme segue. Primeiramente, detectamos os segmentos de linha utilizando a transformada de *Hough*. Após isto, combinamos segmentos de linha de orientação e distância da origem semelhante. Por último, eliminamos linhas que têm ângulo muito diferente de  $0^\circ$  e  $90^\circ$ .

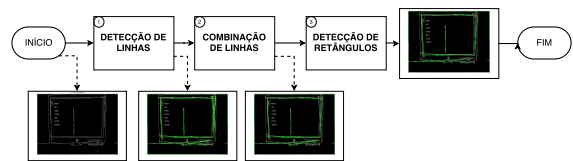


Fig. 3. Diagrama de blocos para detecção de linhas.

Primeiramente, detectamos as linhas através de um algoritmo baseado na transformada de *Hough*, cujos passos serão descritos a seguir:

- Primeiramente, escaneamos toda a imagem de borda e computamos, para cada pixel assinalado como borda, seu conjunto de coordenadas no espaço paramétrico  $(\rho, \theta)$ . Um ponto é acrescentado ao acumulador da coordenada à qual o pixel corresponde, no espaço paramétrico. Assim, quanto mais pixels pertencentes à mesma linha, mais pontos tem o acumulador na coordenada que representa aquela linha;
- Escaneamos o espaço paramétrico para encontrar picos no acumulador, ou seja, as coordenadas que apresentam mais pontos no acumulador. Assinalamos a cada um deles um segmento de linha e guardamos os valores de  $\rho$  e  $\theta$ .

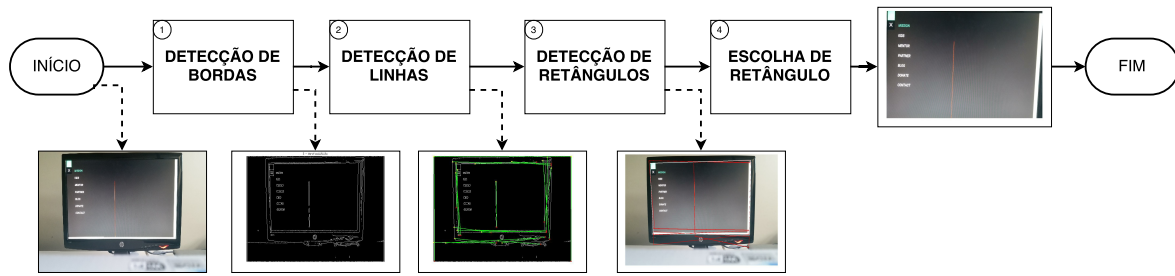


Fig. 1. Diagrama de blocos da metodologia proposta para detecção e extração de telas de TV e monitores.

A fim de evitar linhas pequenas demais, é estabelecido um limiar mínimo de tamanho das linhas;

- Para cada segmento de linha assinalado, procuramos na imagem de borda os pontos que passam por aquele segmento, para assinalar os pontos inicial e final do mesmo;

Devido a diferenças na iluminação, curvatura da lente e ruído, pode haver erro na detecção de linhas executada pelo algoritmo. Assim, um mesmo segmento pode estar sendo identificado como dois ou mais segmentos. A fim de abordar este problema, propomos um esquema de combinação de linhas que combina os segmentos encontrados que obedecem aos seguintes critérios:

- Os segmentos devem ter  $\rho$  e  $\theta$  semelhantes. Devido ao ruído e a problemas na quantização, é possível que algumas linhas do mesmo segmento tenham orientação ou distância da origem um pouco diferentes entre si, portanto estipulamos uma faixa de tolerância para  $\rho$  e  $\theta$ , que compreende 10% a mais ou a menos nos valores de  $\rho$  e  $\theta$ ;
- A distância entre os segmentos de linha não pode ser maior que 5 pixels;
- Segmentos paralelos não podem se sobrepor significativamente quando projetados na direção perpendicular à linha. Neste algoritmo, consideramos uma taxa de sobreposição acima de 60% do tamanho do menor segmento como o critério para eliminar segmentos que se sobrepõem;

Por último, descartamos segmentos de linha que têm ângulos muito diferentes de  $0^\circ$  e  $90^\circ$ . Com isto, diminuímos o número de linhas analisadas nas próximas etapas do método e eliminamos segmentos que podem gerar falsos positivos. Os ângulos foram escolhidos devido as telas (de TVs e monitores) apresentarem ângulos retos.

#### D. Detecção de Retângulos

Os segmentos de linha identificados são rotulados como parte de objetos retangulares através de um modelo baseado no Campo Aleatório de Markov (MRF). Primeiramente, um sistema de vizinhanças é criado para alimentar o modelo com informação espacial sobre os segmentos. A vizinhança de um segmento  $l_i$  é considerado qualquer segmento  $l_j$  que satisfaça as seguintes condições:

- $l_j$  é paralelo ou perpendicular a  $l_i$ ;
- a distância entre  $l_i$  e  $l_j$  não é tão grande nem tão pequena;
- Se  $l_j$  é paralelo a  $l_i$ , eles se sobrepõem significativamente;

Com a informação de vizinhanças, podemos construir um modelo MRF para rotular segmentos de linha como pertencentes à borda de objetos retangulares. Dado um conjunto de segmentos  $L = \{l_1, l_2, \dots, l_n\}$  encontrado nos passos anteriores do algoritmo, e  $F = \{F_1, F_2, \dots, F_n\}$  uma família de variáveis aleatórias, em que cada variável  $F_i$  apresenta valor de 0 a 1 e indica se o segmento  $l_i$  é borda de um objeto retangular ou não. Se  $F_i = 1$ , o segmento  $l_i$  é um lado de retângulo.

O modelo funciona com  $f = \{f_1, f_2, \dots, f_n\}$ , uma configuração de  $F$ , e uma função de energia  $E(f)$  que relaciona a informação espacial provida pelo sistema de vizinhanças e a configuração. A função de energia é minimizada utilizando um algoritmo gradiente descendente, mostrado na equação:

Combinando o modelo e os resultados do sistema, podemos identificar objetos retangulares em uma imagem.

#### E. Escolha do Retângulo que Representa a Tela

A escolha do retângulo que representa a tela da TV/monitor utiliza critérios criados através da análise das propriedades geométricas de uma tela e de características da captura das imagens. Classificamos um retângulo como tela caso ele obedeça às seguintes condições:

- **o retângulo não está nas bordas da imagem:** para evitar que alguma parte da tela não seja capturada. Além disso, o diferencial do método aqui explicado é conseguir capturar a tela de um monitor em um ambiente qualquer;
- **o retângulo possui uma razão de aspecto próxima a 16/9:** a relação de aspecto mais comum em monitores e telas de TV é de 16/9, assim como a razão inversa 9/16 é a mais comum em celulares. Isso se deve em parte por ser o formato padrão de imagens em *HDTV*, *Full HD* e transmissão de TV digital e analógica [19]. Assim, a fim de evitar falsos positivos, o método descarta, dentre os objetos encontrados, os que não tiverem uma relação de aspecto próxima dos monitores que estão sendo estudados;
- **o retângulo preenche mais que 30% da imagem:** é considerado que a tela, embora não possa ser o único objeto presente na imagem, precisa ser grande o suficiente para que seja possível extrair informações dela. Para isso, é necessário que a tela ocupe um espaço mínimo da imagem. Assim, a área dos objetos retangulares é calculada e comparada com o tamanho da imagem, a fim de descartar os que têm área abaixo da mínima;

O resultado da seleção deve ser apenas retângulos de tamanho considerável e com a relação de aspecto próxima

à de TVs e monitores. Com isto, se faz necessário um critério de verificação para que o retângulo que mais se assemelha à tela seja o escolhido como verdadeiro. Neste artigo, utilizamos os seguintes critérios [11]:

- caso não seja encontrado retângulo atendendo às condições, a resposta do método é que nenhum retângulo que representa a tela foi encontrado;
- caso seja encontrado somente um retângulo atendendo às condições, este retângulo é considerado a tela;
- caso seja encontrado mais de um retângulo atendendo às condições, os dois maiores retângulos encontrados ao final do processo representam as bordas da tela e do monitor. Portanto, o segundo maior retângulo é escolhido como tela;

### III. PROCEDIMENTOS EXPERIMENTAIS

A fim de validar o método de detecção do conteúdo de telas de TV e monitores, foram feitos experimentos que abordam o problema da inspeção automática de TVs. Os algoritmos dos experimentos foram implementados na linguagem MATLAB, na versão R2012a, em ambiente Linux, processador Intel(R) Core(TM) i7 CPU 3,40GHz e memória RAM 16GB.

#### A. Base de dados

Para os experimentos descritos nesta seção, criamos duas bases de dados com fotos de aparelhos de TV e monitores. Estas bases foram criadas com o objetivo de apresentar um conjunto de imagens de telas com conteúdo em alta resolução e diferentes condições de iluminação, pose e distância, para tornar possível a realização de experimentos que verifiquem o desempenho da metodologia sob diversas condições de entrada.

A primeira base é formada por 333 imagens de teste e 6 imagens de referência. As imagens de teste apresentam fotos coloridas de aparelhos de TV e monitores sob condições não controladas de iluminação, background e distância entre câmera e aparelho retratado, divididas igualmente entre três resoluções diferentes:  $5312 \times 2988$ ,  $3264 \times 2448$  e  $2048 \times 1152$  pixels. As imagens de referência são coloridas e têm resolução de  $1600 \times 900$  pixels. Nestas imagens são representados os conteúdos apresentados nas telas de TV e monitores nas imagens de teste. As imagens de teste são rotuladas com quatro coordenadas que representam os limites da tela.

A segunda base é formada por 230 imagens de teste e 40 imagens de referência. As imagens de teste apresentam fotos coloridas de monitores, diferenciadas das fotos da primeira base por estarem sob condições controladas de iluminação, background e distância entre câmera e aparelho retratado. Todas as imagens de teste têm resolução de  $5312 \times 2988$  pixels. As imagens de referência são coloridas e representam os conteúdos apresentados nas telas dos monitores nas imagens de teste. Assim como na primeira base, as imagens de teste são rotuladas com quatro coordenadas que representam os limites da tela.

#### B. Métricas Utilizadas

Neste artigo, as métricas utilizadas foram:

- LAE: o método *least average error* (LAE) mede a diferença entre duas imagens. A fim de eliminar erros

devidos a diferenças de iluminação, as imagens são normalizadas antes da comparação.

- NCC: o método *normalised cross-correlation* (NCC), calcula a similaridade entre duas imagens através da correlação cruzada. Assim como no LAE, a fim de eliminar erros devidos a diferenças de iluminação, as imagens são normalizadas antes do cálculo.
- NCC-BB: este método calcula a similaridade entre duas imagens utilizando a mesma correlação cruzada do método NCC. Entretanto, neste método cada imagem é separada em blocos. A autocorrelação é calculada em cada bloco e, após isto, a correlação cruzada entre os blocos das diferentes imagens é calculada. A diferença entre as duas é calculada e somada bloco a bloco para obter a pontuação final.
- Detecção: para cada imagem de saída do sistema, comparamos o conjunto de coordenadas dos vértices do retângulo considerado tela com o conjunto de coordenadas dos vértices rotulados manualmente. Conforme uma tolerância, a classificação é considerada sucesso, caso contrário considera-se fracasso na detecção.

#### C. Resultados: Primeiro Experimento

No primeiro experimento, utilizamos a metodologia para detectar um retângulo considerado tela em cada imagem de entrada. Como entrada, utilizamos todas as imagens de teste das duas bases de dados, totalizando 1104 imagens. O objetivo deste experimento é comparar a detecção em ambiente controlado e não controlado. A Tabela I apresenta a detecção, segundo a definição apresentada na Seção III-B, das imagens da primeira e da segunda base. Para fins de comparação, as mesmas imagens foram submetidas ao algoritmo de detecção de retângulos descrito em [2] e para este caso o algoritmo não detecta nenhum retângulo, acreditamos que isso é devido às imagens apresentarem retângulos não triviais para detecção.

Método	Base	Imagens	Telas Detectadas	Taxa de Detecção
<b>Proposto</b>	1	333	42	12,61%
Método [2]	1	333	0	00,00%
<b>Proposto</b>	2	230	110	47,83%
Método [2]	2	230	0	00,00%

TABELA I

TAXA DE DETECÇÃO NAS DUAS BASES DE DADOS.

#### D. Resultados: Segundo Experimento

No segundo experimento, utilizamos a metodologia para detectar o conteúdo da tela em cada imagem de entrada e classificamos a imagem comparando este conteúdo com imagens de referência. Como entrada, utilizamos as imagens de teste da primeira base de dados que representam menus, assim como imagens de referência que mostram seleções de menu. O objetivo deste experimento é verificar o desempenho da metodologia em um sistema de comparação de imagens. Após detectar a tela, comparamos a imagem com diversas imagens de referência e atribuímos pontuações a cada comparação, segundo as métricas LAE, NCC e NCC-BB, definidas na Seção III-B. Para este experimento, utilizamos quatro classes que correspondem ao conteúdo presente na tela nas imagens

de teste. Ele corresponde a quatro seleções possíveis de um mesmo menu, como mostrado na Figura 4. As imagens têm o mesmo tamanho e a mesma cor. A única diferença entre elas é a coloração que corresponde à seleção naquela classe. Para a imagem ser considerada correta, o sistema deve atribuir à imagem a classe que corresponde à seleção de menu que ela apresenta.

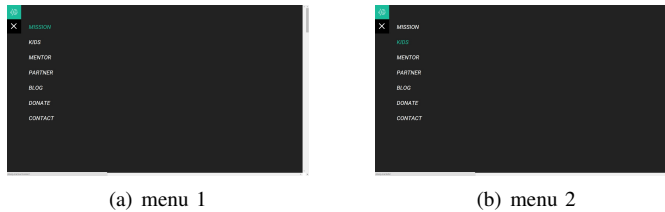


Fig. 4. Conjunto de dois menus, de quatro disponíveis, utilizados como conteúdo das telas de TV.

O sistema tem como saída, para cada imagem de teste, um conjunto pontuações, conforme descrito na Seção III-B, referentes à comparação entre o conteúdo da tela e as imagens de referência. A Tabela II mostra as pontuações calculadas na comparação de uma imagem com quatro imagens de referência:

Referência	LAE	NCC	NCC-BB
menu 1	4.5160 $e^6$	-0.25312	-3.2196
menu 2	4.5186 $e^6$	-0.25957	-3.2294
menu 3	4.5162 $e^6$	-0.2539	-3.226
menu 4	4.5167 $e^6$	-0.2553	-3.2292

TABELA II

PONTUAÇÕES DA COMPARAÇÃO ENTRE UMA IMAGEM REPRESENTANDO O MENU 1 E QUATRO IMAGENS DE REFERÊNCIA.

Estas pontuações são comparadas entre si a fim de classificar a imagem. O rótulo dado à imagem pelo sistema é comparado com a classe com a qual cada imagem é rotulada previamente. Caso o rótulo corresponda à classe, a imagem é considerada correta.

#### IV. CONCLUSÕES

Neste artigo, apresentamos uma metodologia de detecção de telas de TV e monitores e análise do conteúdo extraído que utiliza um modelo baseado no campo aleatório de Markov (MRF). A metodologia proposta, que é uma alternativa para o problema de detecção de objetos retangulares, envolve diversas etapas, tais como: extração de bordas por análise de gradiente multi-dimensional; detecção de segmentos de linha utilizando a transformada de *Hough* e classificação de segmentos de linha em objetos retangulares. As simulações consideraram dois experimentos, um para comparação da tela detectada e outro para análise de conteúdo dos menus. No primeiro caso, o método proposto possui desempenho satisfatório e quando comparado a um método similar percebe-se que não há detecção de nenhum retângulo. No segundo experimentos podemos notar que o método pode ser utilizado para análise de conteúdos que envolvem retângulos tais como os que envolvem *menus*.

#### AGRADECIMENTOS

Parte dos resultados apresentados neste trabalho foram obtidos através do Projeto de Pesquisa e formação de recursos humanos, em nível de graduação e pós-graduação, nas áreas de automação industrial, software para dispositivos móveis e TV Digital, financiado pela Samsung Eletrônica da Amazônia Ltda., no âmbito da Lei no. 8.387 (art. 2º)/91. Esta pesquisa também foi financiada com recursos da CAPES/CNPq e da Fundação de Amparo à Pesquisa do Estado do Amazonas (FAPEAM).

#### REFERÊNCIAS

- [1] Y. Liu, T. Ikenaga, and S. Goto, "An mrf model-based approach to the detection of rectangular shape objects in color images," *Signal Processing*, vol. 87, no. 11, pp. 2649–2658, 2007.
- [2] C. R. Jung and R. Schramm, "Rectangle detection based on a windowed hough transform," in *Proceedings of the 17th Brazilian Symposium on Computer Graphics and Image Processing*. IEEE, 2004, pp. 113–120.
- [3] Y. Zhu, B. Carragher, F. Mouche, and C. S. Potter, "Automatic particle detection through efficient hough transforms," *IEEE Transactions on Medical Imaging*, vol. 22, no. 9, pp. 1053–1062, 2003.
- [4] D. Lagunovsky and S. Ablameyko, "Straight-line-based primitive extraction in grey-scale object recognition," *Pattern Recognition Letters*, vol. 20, no. 10, pp. 1005–1014, 1999.
- [5] T. Wen-bing, T. Jin-wen, and L. Jian, "A new approach to extract rectangular building from aerial urban images," in *Proceedings of the International Conference on Signal Processing*, vol. 1. IEEE, 2002, pp. 143–146.
- [6] G. Roth and M. D. Levine, "Extracting geometric primitives," *CVGIP: Image Understanding*, vol. 58, no. 1, pp. 1–22, 1993.
- [7] —, "Geometric primitive extraction using a genetic algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 901–905, 1994.
- [8] E. Lutton and P. Martinez, "A genetic algorithm for the detection of 2d geometric primitives in images," in *Proceedings of the International Conference on Pattern Recognition, Computer Vision and Image Processing*, vol. 1. IEEE, 1994, pp. 526–528.
- [9] G. Wang, F. Miao, W. Zhang, and H. Yu, "A dynamic and interactive diagnosing and testing method for development of digital tv receiver system," in *Proceedings of the International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, vol. 3. IEEE, 2007, pp. 1107–1112.
- [10] A. Sung and B. Choi, "An interaction testing technique between hardware and software in embedded systems," in *Proceedings of the Software Engineering Conference*, 2002, pp. 457–464.
- [11] I. Kastelan, M. Katona, D. Marijan, and J. Zloh, "Automated optical inspection system for digital tv sets," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 1, pp. 1–17, 2011.
- [12] A. N. Rau, "Automated test system for digital tv receivers," in *Proceedings of the International Conference on Consumer Electronic*. IEEE, 2000, pp. 228–229.
- [13] A. Rama, R. Alujas, and F. Tarres, "Fast and robust graphic character verification system for tv sets," in *Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services*, June 2007, pp. 19–19.
- [14] N. Vercammen, N. Staelens, A. Rombaut, B. Vermeulen, and P. De-meester, "Extensive video quality evaluation: A scalable video testing platform," in *Proceedings of the International Conference on Computer and Information Technology*, Dec 2008, pp. 91–97.
- [15] D. Marijan, V. Zlokolica, N. Teslic, V. Pekovic, and T. Tekcan, "Automatic functional tv set failure detection system," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 1, pp. 125–133, 2010.
- [16] M. Katona, I. Kastelan, V. Pekovic, N. Teslic, and T. Tekcan, "Automatic black box testing of television systems on the final production line," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 1, pp. 224–231, 2011.
- [17] H.-C. Lee and D. R. Cok, "Detecting boundaries in a vector field," *IEEE Transactions on Signal Processing*, vol. 39, no. 5, pp. 1181–1194, 1991.
- [18] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.
- [19] E. T. S. I. (ETSI), *Digital Video Broadcasting (DVB) Specification for the use of Video and Audio Coding in Broadcasting Applications based on the MPEG-2 Transport Stream*, TS ETSI TS 101 154, 2012.