

"DETECÇÃO DE PADRÕES DE LEGENDAS EM IMAGENS DE RITMO VISUAL A PARTIR DO
DETECTOR DE HARRIS"?

Relatório do segundo de MC920

Aluno: Carlos Eduardo Rosa Machado **RA:** 059582

Aluno: Tiago Chedraoui Silva **RA:** 082941

Aluno: William Marques Dias **RA:** 065106

Resumo

A consistência de um filtro de bordas é de suma importância para interpretações de sequências de imagens 3D para recursos que utilizam algoritmos rastreamento.

Para abranger as regiões da imagem que contém textura e características isoladas, uma combinação de detector de bordas e cantos baseados na função de auto-correlação local é utilizado.

A partir do detector de Harris, avaliou-se a detecção de padrões de legendas em imagens de ritmo visual.

Sumário

1	Introdução	1
2	Métodos	1
3	Detecção de contornos em imagens	2
3.1	Resultados	2
4	Conclusão	4

1 Introdução

Baseado no detector de cantos de Moravec, Chris Harris e Mikes Stephens [1] desenvolveram melhorias no algoritmo e implementaram-na de modo que parefeçoaram a detecção de cantos.

Nesse laboratório utilizamos tais melhorias para avaliar sua aplicação em detecção de legendas.

2 Métodos

Desenvolveu-se em python [2] um programa [4] para aplicar o detector de cantos de Moravec.

Dado uma imagem I retorna-se a imagem com os cantos realçados. Para isso aplica-se a fórmula:

$$E_{x,y} = \sum_{u,v} w_{u,v} |I_{x+u,y+v} - I_{u,v}|^2 \quad (1)$$

No entanto, o operador de Moravec sofre de alguns problemas cujas soluções são apresentadas no paper de Chris Harris e Mikes Stephens [1]. Abaixo estão listadas as que serviram de base para uma implementação em nossa pesquisa.

1. **A resposta é anisotrópica, porque somente um conjunto discreto de deslocamentos a cada 45 graus é considerado** - Todos os pequenos deslocamentos são cobertos realizando uma expansão analítica sobre a origem do deslocamento. Assim:

$$E_{x,y} = \sum_{u,v} w_{u,v} |xX + yY + O(x^2, x^2)|^2 \quad (2)$$

Em que:

$$X = 1 \otimes (-1, 0, 1) = \delta I \delta x$$

$$Y = 1 \otimes (-1, 0, 1)^T = \delta I \delta y$$

Para pequenos deslocamentos, E pode ser escrito como:

$$E_{x,y} = Ax^2 + 2Cxy + By^2 \quad (3)$$

Em que

$$A = X^2 \otimes w$$

$$B = Y^2 \otimes w$$

$$C = (XY) \otimes w$$

2. **A resposta é ruidosa devido ao fato de a janela ser binária e retangular** - Usar uma janela suave circular, como por exemplo uma Gaussiana.

$$w_{x,y} = \exp(-(u^2 + v^2)/2\sigma^2) \quad (4)$$

3. **O operador responde muito rapidamente às bordas porque somente o mínimo de E é levado em conta** - reformular a medida do canto para fazer uso da variação de E com a direção da mudança.

A mudança, E para o pequeno deslocamento (x, y) pode ser escrita como:

$$E_{x,y} = (x,y)M(x,y)^T \quad (5)$$

Em que a Matriz simétrica 2x2 é dada por:

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix}$$

Usando a matrix M calculamos:

$$Tr(M) = A + B$$

$$Det(M) = AB - C^2$$

Para realizar um avaliar os cantos, fazemos:

$$R = Det - kTr^2 \quad (6)$$

Em que, se $R > 0$, temos uma região de canto, se $R < 0$, é uma região de borda e, se $R \approx 0$, temos uma região plana.

3 Detecção de contornos em imagens

Aplicando as melhorias propostas no paper de Harris e Stephens, fez-se uma sequência de experimentos em imagens de ritmo visual, que podem ser encontradas no site da disciplina [3].

3.1 Resultados

A partir do detector de Harris, conseguiu-se para um alto limiar, possuir, na maioria das vezes, todos os padrões de legendas. Alguns pontos, apesar de estarem fora de uma possível legenda, por estarem isolados de qualquer outro ponto, podem ser facilmente descartados. Abaixo encontram-se diversos resultados nos quais os pontos de contorno podem ser vistos. Devido a mudança somente de um pixel, é aconselhado que a imagem seja aproximada, as imagens podem ser encontradas no repositório do grupo [5].

Para a imagem do filme Harry Potter com um limiar, que é dependente de nossa implementação (ou seja, dependendo de cada implementação o valor de limiar pode ser maior ou menor) próximo a 7500000000, conseguimos identificar a maioria das legendas.

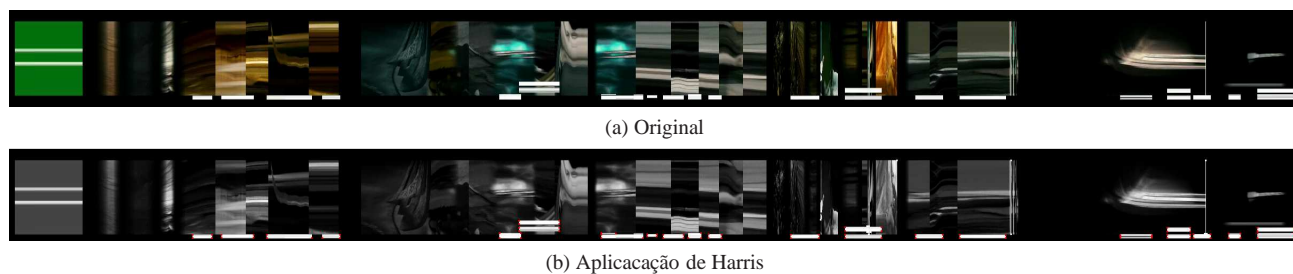


Figura 1: Encontrando legendas - pontos de contornos das legendas realçados em vermelho

Vê-se, também, que o método proposto por Harris e Stephens é superior ao de Moravec para o objetivo desse estudo. Para a imagem `commercial_1.png`, utilizando Moravec, o máximo de cantos de legenda encontrados sem muitos pontos desinteressantes foi o seguinte:

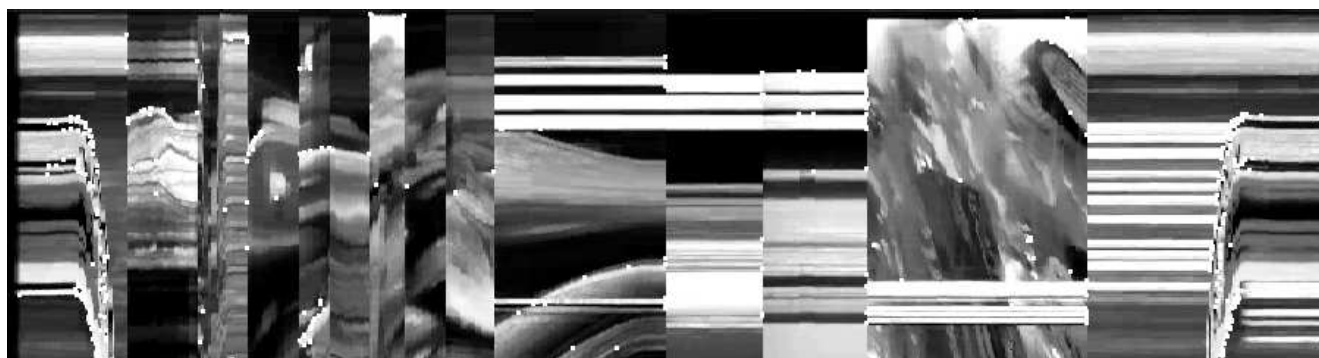


Figura 2: Detector de Moravec

Já com o detector de Harris, obtivemos:

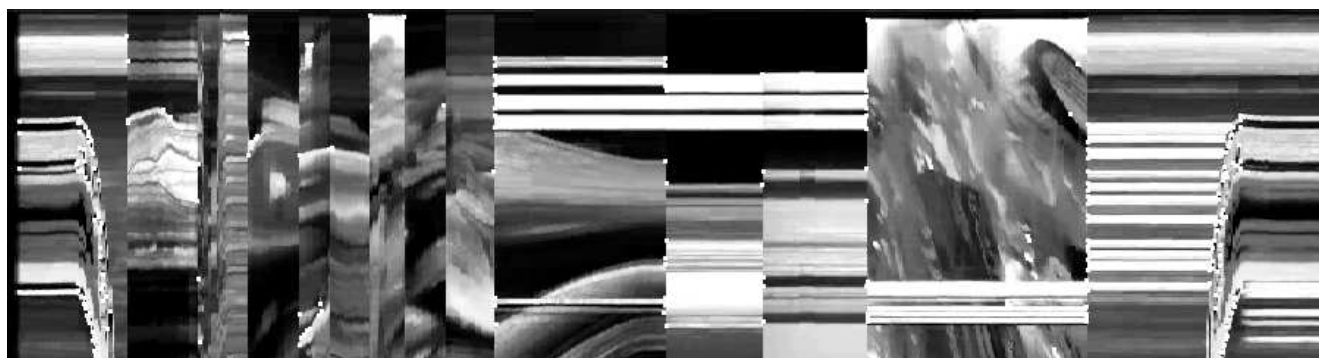


Figura 3: Detector de Harris

Podemos facilmente ver que há uma precisão maior na detecção de legendas nesta última imagem. Além disso, a quantidade de pontos que não fazem parte de um "retângulo de legenda" é bem menor.

De maneira aproximada, implementamos um algoritmo que procurava por pontos que poderiam formar um retângulo. Como saída, temos a figura original com retângulo de contorno de linhas vermelhas. Para a figura harrypotter_1.png, a saída pode ser vista abaixo.

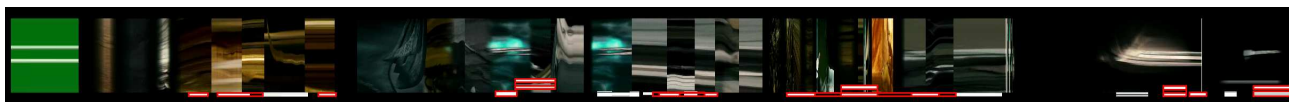


Figura 4: Detector de Harris

Conseguiu-se portanto identificar um grande número de legendas e, a partir de um algoritmo mais preciso, poderá se obter a maioria das legendas das imagens.

4 Conclusão

Pode-se dizer que o detector de contornos de Harris pode ser usado para a detecção de padrões de legendas em imagens de ritmo visual a partir do detector de Harris. Dependendo do limiar, encontra-se todas as legendas, com alguns de pontos fora das curvas das legendas, mas que são indetectáveis e descartáveis. Criando-se uma possível limitação na área de busca de uma imagem, e possuindo dados precisos como a maior área e a menor de uma legenda, pode-se criar um filtro com uma boa precisão.

Referências

- [1] Chris Harris e Mike Stephens *A COMBINED CORNER AND EDGE DETECTOR*. Disponível em <http://www.ic.unicamp.br/neucimar/cursos/MO443/2011-s01/tp2.zip>, [Último acesso: 17/04/2011].
- [2] *Python Programming Language – Official Website*. Disponível em <http://www.python.org/>.
- [3] *Imagens para avaliação*. Disponível em <http://www.ic.unicamp.br/neucimar/cursos/MO443/2011-s01/tp2.zip>, [Último acesso: 17/04/2011].
- [4] *Código do projeto*. Disponível em <https://github.com/tiagochst/mc920-lab1/tree/master/lab2>.
- [5] *Imagens do relatório*. Disponível em <https://github.com/tiagochst/mc920-lab1/tree/master/lab2/img>.