**Resumo:**

O presente trabalho demonstra de forma prática a aplicação do cálculo de derivadas na área da Engenharia da Computação. Baseado em um segmento de grande importância no desenvolvimento de tecnologias, este projeto visou estudar o processamento de imagens, mais especificamente a detecção de bordas proposta por John Canny, para demonstrar aos alunos que cursam o primeiro ou segundo período a importância do estudo da diferenciação durante a formação. Espera-se que estudantes que já desenvolveram uma noção do cálculo possam compreender de forma simples a área abordada. O processo para extração de bordas conhecido como método Canny, utiliza a variação da intensidade do pixel de uma imagem para gerar uma imagem composta apenas por bordas. Em complemento à exposição dos passos envolvidos nesse procedimento, serão apresentados os resultados de um algorítimo desenvolvido para aplicar a imagem resultante do método de detecção de bordas para realizar medições de um objeto em tempo real. Este projeto proporcionou aos desenvolvedores uma visualização prática da real aplicação do cálculo de derivadas. Ademais, foi possível conhecer uma área onde há muito a se pesquisar e a se desenvolver em novos projetos.

**Palavras-chave:** Cálculo diferencial; Processamento de imagens; Detecção de bordas; Método Canny; Medição de objetos.

**Abstract**:

The current paper demonstrates a practical application of the derivative calculus in the Computer Engineering area. Based on a segment of great importance in the development of technologies, this projects aimed at studying image processing, more precisely the edge detection proposed by John Canny, to demonstrate to students who are in the firsts period the importance of learning differentiation during the formation. It is hoped that students who have already developed a notion of calculus can understand in a simple way the area approached. The process for edge extraction, known as the Canny method, uses the variation of the pixel intensity in an image to generate an image composed only of edges. In addition of exposing the process involved in this procedure, the results of an algorithm developed to apply the image resulted from the edge detection method to perform measurements of an object in real time will be presented. This project gave developers a hands-on look at the actual application of derivative calculus. Moreover, it made possible knowing an area where there are a many researches and developments to be applied in new projects.

Key-words: Derivative calculus; Image processing; Edge detection; Canny method; Object measurement.

# Introdução

## Problematização

A derivada tem como conceito teórico fornecer a variação instantânea de uma função em relação às variáveis em um dado momento. Geometricamente a derivada fornece uma reta tangente, tal que seu ângulo indica o quanto o ponto está variando em relação aos seus vizinhos.

Mesmo com um conceito muito importante, aplicar derivada em situações reais não é algo trivial. Normalmente as situações reais possuem diversas variáveis que impossibilitam chegar a uma fórmula simplificada como é visto em sala de aula nos primeiros períodos da engenharia. Isto significa que existe uma dificuldade em demonstrar a aplicação do cálculo de derivada na área de computação para os alunos do primeiro ano em Engenharia de Computação da UFOP. Entretanto, existem algumas áreas em que é possível ver a aplicação da derivada e ter uma base sobre os mecanismos computacionais que a operam.

A área a ser abordada deve seguir os seguintes parâmetros:

* Não exigir conhecimento que ultrapasse a grade curricular do primeiro e segundo período;
* Ser visível sua aplicação;
* Não demandar uma quantidade grande de tempo para sua execução.

Portanto, faz se necessário responder: Qual área na computação é aplicada os conceitos de derivada de forma que os alunos possam compreender de forma intuitiva? Esse trabalho propõem o processamento de imagem, sobretudo a detecção de borda, como resposta para essa questão.

## Justificativa

O processamento de uma imagem tende a ser um procedimento de entrada e saída de dados que consiste na transformação de uma imagem realçando os fatores de interesse. Por trás desse processo ocorrem vários outros que não podem ser visualizados em um primeiro instante sem que seja feito um estudo sobre o tema, um desses processos é a derivada, onde muitos alunos a utilizam somente para fins matemáticos e não sabem sobre sua importância em diversos seguimentos.

A detecção de borda é uma das vertentes do processamento de imagem, sendo essa a área a ser trabalhada no desenvolvimento desse projeto. Na internet existem diversas bibliotecas que podem ser implementadas em algoritmos sem que o aluno desenvolva técnicas muito avançadas na programação. Ademais, o método Canny é um conceito consolidado, por isso softwares que trabalham com o processamento de imagens já possuem funções próprias para a sua aplicação, o que demanda menor tempo na elaboração do algoritmo. Outra razão da escolha desse processo explica-se pelo fato de as etapas que envolvem a detecção de borda e sua aplicação serem de fácil percepção pelos alunos. Além disso, os conceitos aplicados não exigem conhecimento que ultrapasse o que e demandado no primeiro e segundo período.

## Objetivo

Esse trabalho tem como objetivo geral compreender a aplicação de derivada na Engenharia da Computação. Sobretudo, será abordado a área de processamento de imagens e extração de padrões com foco na detecção de borda utilizando o método Canny. Esse objetivo geral pode ser detalhado nos seguintes objetivos específicos:

* Compreender a utilização do método Canny para a detecção de borda;
* Compreender o cálculo da derivada aplicada no método;
* Desenvolver um algoritmo que detecta bordas em uma imagem;
* Aprimorar o algoritmo para realizar medições de objetos;
* Implementar o algoritmo para medir objetos em tempo real.

# Aplicabilidade do cálculo na engenharia

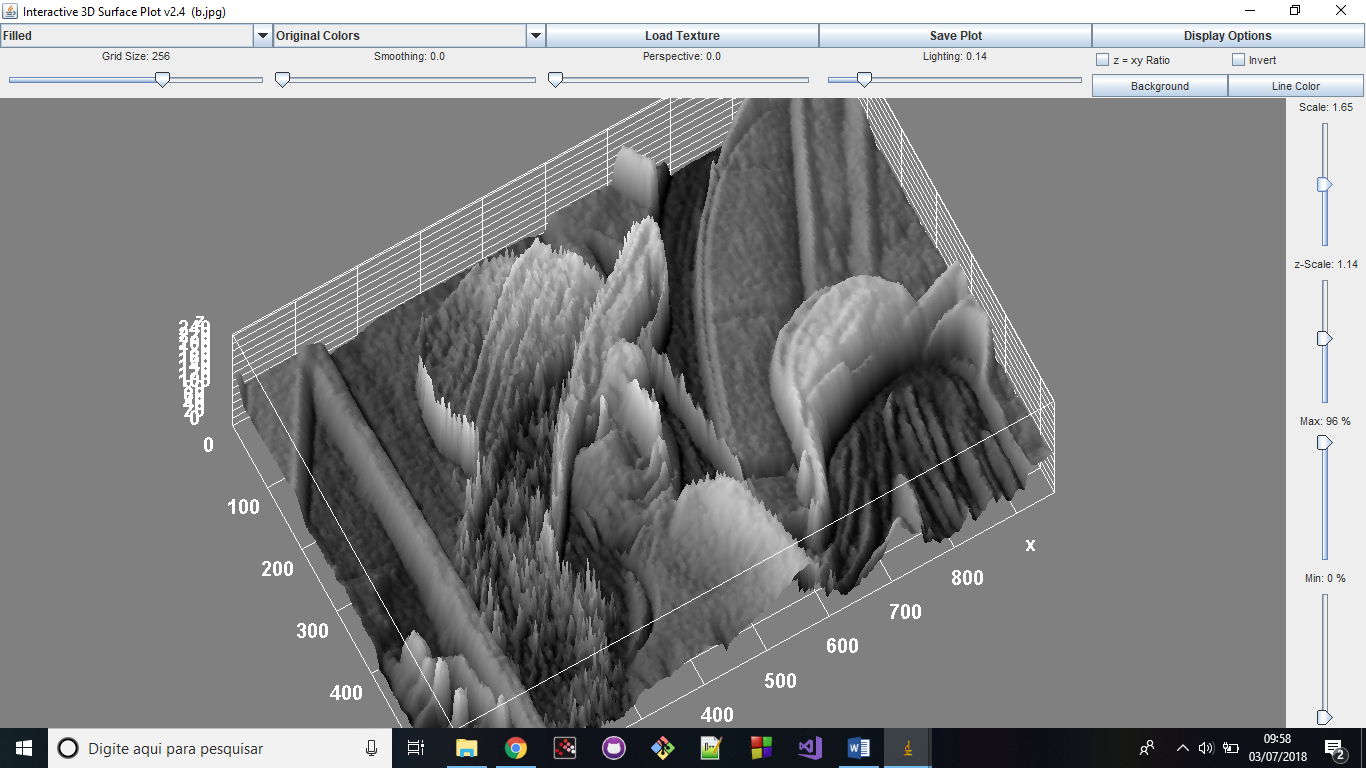
# Referencial teórico

Para o entendimento desse projeto é necessário revisar alguns conceitos na área da computação, sobretudo o processamento de imagem com foco na detecção de borda. Dessa forma, pretende-se introduzir os conceitos sobre o processamento de imagem, apresentando como é armazenada uma imagem no computador e o método utilizado pelo grupo na detecção de borda. Por fim, são exibidos os conceitos sobre programação e sobre a biblioteca utilizada para desenvolver o algoritmo do projeto.

# Processamento de imagem

O processamento de imagem atua na influência da iluminação refletida pela superfície de corpos com o objetivo de fazer análises sobre esse. Utilizando imagens é possível extrair informações sobre as características físicas e geométricas básicas dos objetos, tais como a dimensão, a área, o perímetro e a forma. O processamento de imagem é utilizado para extrair essas características e assim encontrar padrões que possibilitem maior conhecimento sobre o corpo.

Segundo introdução a computação gráfica (p. 8) uma imagem monocromática, isto é que possui uma única cor, é uma função contínua bidimensional dada por F (x, y), tal que x e y são as coordenadas espaciais dos pontos da fotografia e F representa a intensidade da luz no local. Para um computador processar imagens é necessário armazenar cada ponto, denominados como *Picture element (pixel)*, dessas em um *array* bidimensional, ou seja, em uma matriz, por meio de um processo conhecido como amostragem (segmentação por detecção de borda p 21). Esse tipo de armazenamento tem como vantagem a possibilidade de aplicar operações matriciais sobre a fotografia, como soma, multiplicação por escalar e etc. Nesse caso, os pixels possuem representações no R3: as coordenadas x e y que correspondem sua posição na imagem, e a intensidade de cor no eixo z, veja a imagem.



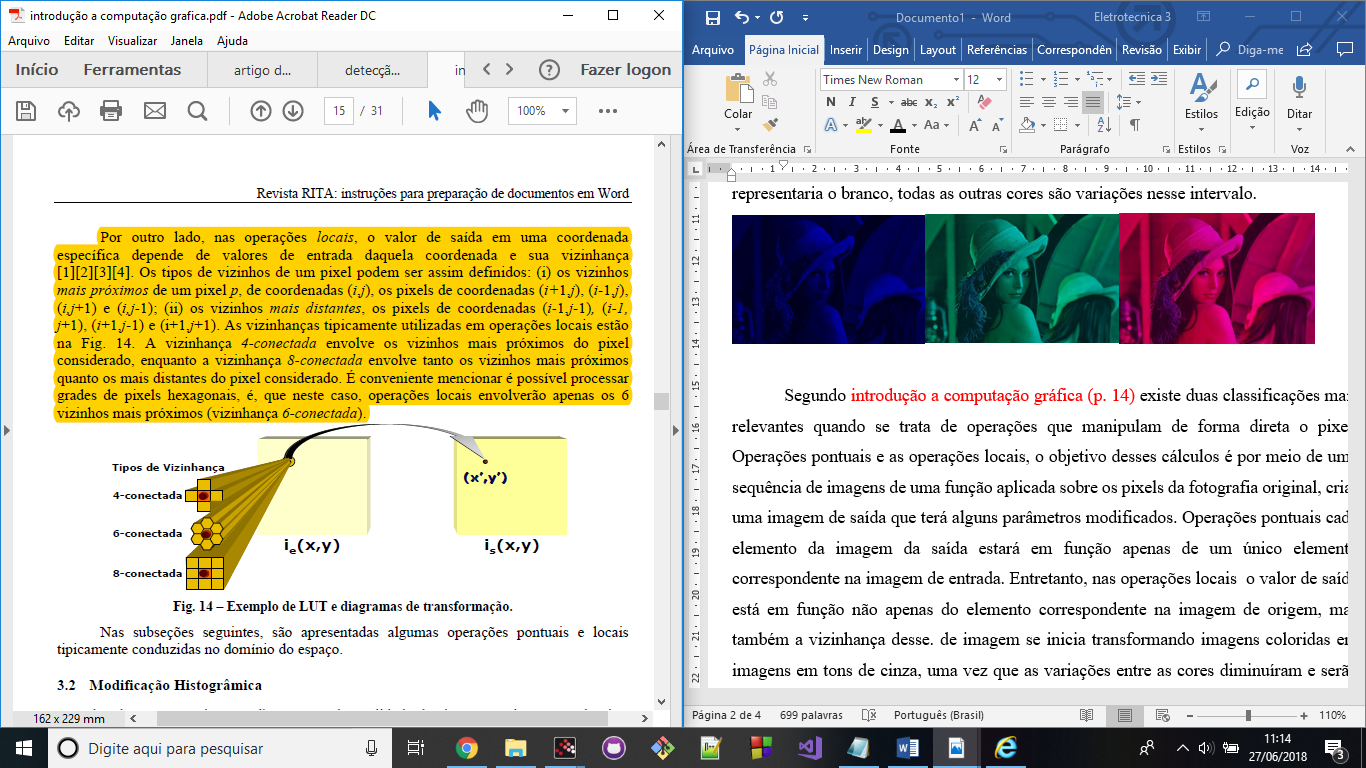
Normalmente as imagens são utilizadas no sistema *Red, Green and Blue (RGB)*, isso significa que cada pixel é uma aproximação resultante das intensidades de vermelho, azul e verde. A união das três cores no pixel forma a imagem de 3 banda, conhecida como imagem colorida segmentação por detecção de borda (p 11).

F (x, y) = Fr (x, y) + Fg (x, y) + Fb (x, y) (fonte introdução a computação gráfica p 8)



Majoritariamente as imagens utilizadas no trabalho foram de 8 bits, ou seja, elas suportam tonalidades de cores nas escalas RGB que variam de 0 a 255, totalizando 2^8 tipos de cores. Segundo segmentação processamento (p 21) O tamanho de um arquivo digital está diretamente relacionado com a qualidade dos detalhes visíveis e o tamanho do intervalo de cores, entretanto é importante ressaltar que o processamento de arquivos grandes demanda maior tempo, podendo ser inviável sua aplicação. Por convenção na área de computação visual adotou-se que a intensidade do pixel igual a 0 representaria o preto e o 255 representaria o branco, todas as outras cores são variações nesse intervalo.

Segundo introdução a computação gráfica (p. 14) existem duas classificações mais relevantes quando se trata de operações que manipulam de forma direta o pixel: as operações pontuais e as operações locais ou por máscaras. O objetivo desses cálculos é, por meio de uma sequência funções aplicadas sobre os pixels da fotografia original, criar uma imagem de saída que terá alguns parâmetros modificados. Nas operações pontuais cada elemento da imagem da saída estará em função apenas de um único elemento correspondente na imagem de entrada. Entretanto, nas operações locais o valor de saída é dependente não apenas do elemento correspondente na imagem de origem, mas também a vizinhança desse pixel, isto é, os elementos que estão dentro da máscara. As máscaras são matrizes com dimensões pequenas que podem ser aplicadas em cada pixel da imagem segmentação por detecção de borda. A imagem a seguir exemplifica essa operação:



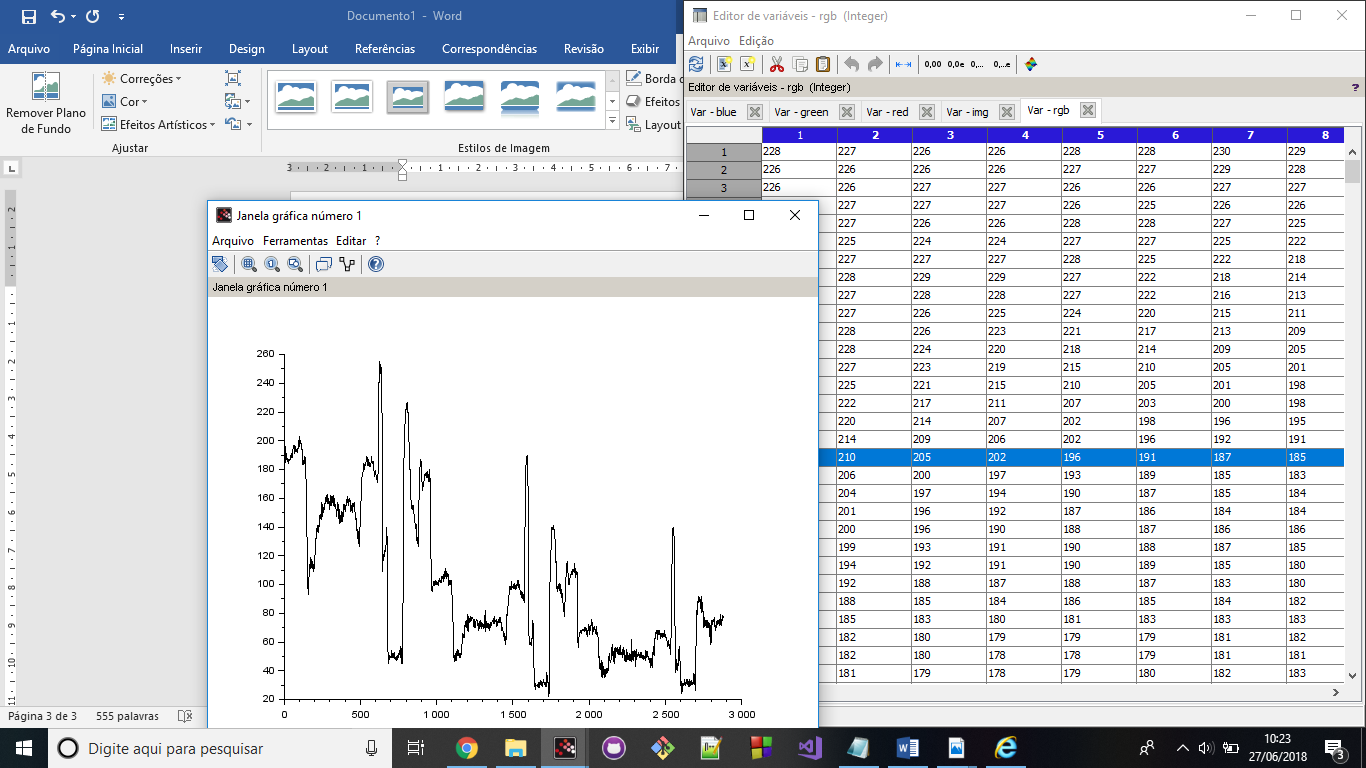
(Fonte: introdução a computação gráfica (p. 15))

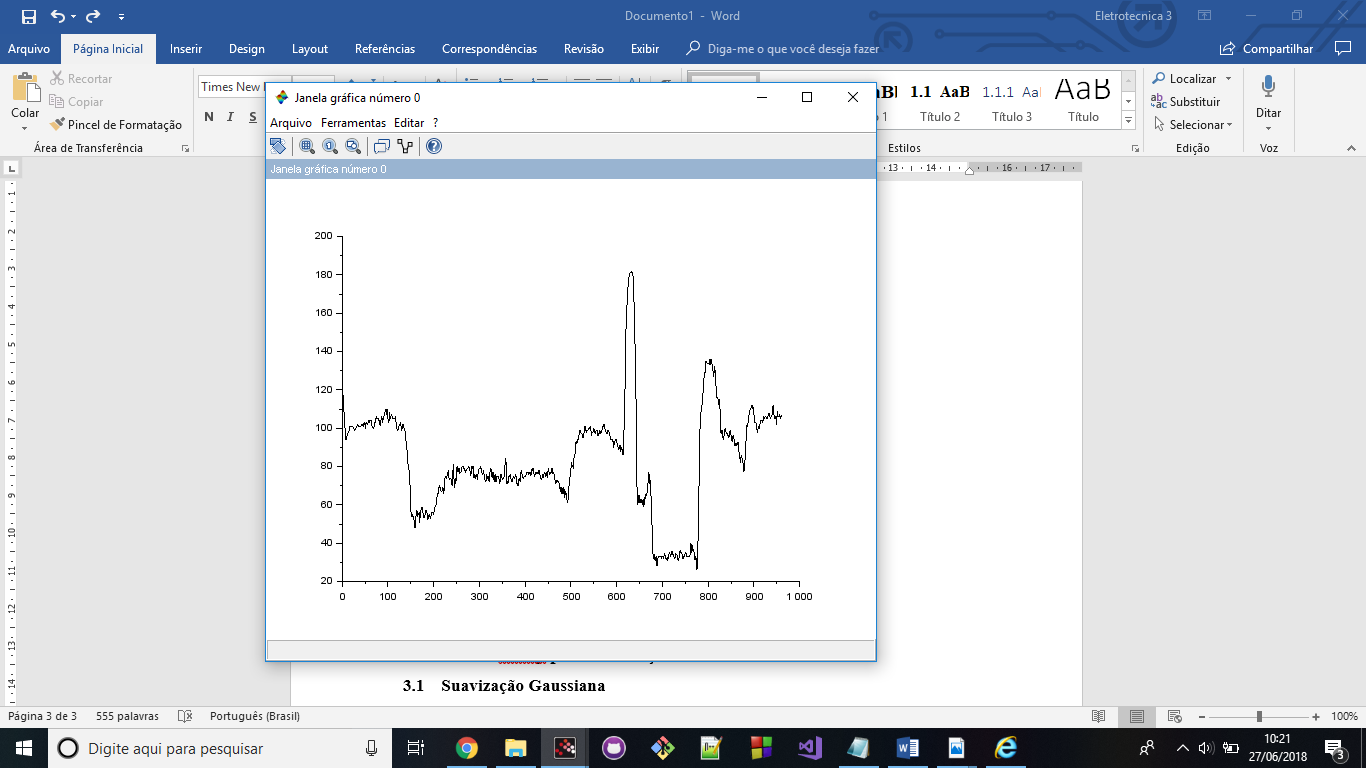
Na aplicação da máscara o pixel a ser tratado é posicionado no centro da matriz de operação e o resultado se torna um novo pixel na mesma posição.

A par desses conceitos, o processamento da imagem se inicia transformando imagens coloridas em imagens em tons de cinza, uma vez que as variações entre as cores diminuirão e será mais fácil aplicar outras operações sobre a fotografia. Segundo o site oficial do *Matlab (https://www.mathworks.com/help/matlab/ref/rgb2gray.html)* a transformação que a biblioteca incluída no algoritmo aplica é baseada em uma média ponderada das componentes RGB de cada pixel, dada pela formula a seguir:

F (x, y) = (0,299 \* Fr(x,y)) + (0,587 \* Fg(x,y)) + (0,114 \* Fb(x,y));

É possível observar que essa transformação pode ser definida como uma operação pontual e a diminuição nas variações da intensidade de cor após a transformação em escala de cinza pode ser vista nas imagens a seguir, de forma que o gráfico representa a intensidade da cor do pixel em função da posição do mesmo no eixo horizontal.





Após a transformação em escalas de cinza inicia-se a aplicação dos métodos para a detecção de borda, o grupo escolheu o método criado por John Canny uma vez que os parâmetros ótimos desse processo supriam as necessidades do nosso objetivo. Esses parâmetros serão melhor detalhados nos tópicos seguintes. A imagem que o Método resulta na saída é denominada imagem binária, isso porque os elementos variam entre 0 e 1, de forma que o 1 representa que naquele espaço há uma borda.



## Método canny para a detecção de borda

Canny apresentou a primeira ideia sobre detecção de borda em 1983 em seu artigo denominado *A variational approach to edge detection*. Nesse artigo são apresentados três parâmetros que Canny considera como sendo um filtro ideal para a detecção de borda. (1) O filtro deveria ter a menor taxa de erro possível, de forma a retornar a maior quantidade de bordas verdadeiras de uma imagem, considerando verdade uma borda proveniente da variação de cor entre um objeto e um segundo plano. Além disso, (2) a posição dos pixels marcados com variações de cores deve estar ao centro da borda. Por fim, (3) a borda deveria ter como espessura o tamanho de um pixel.

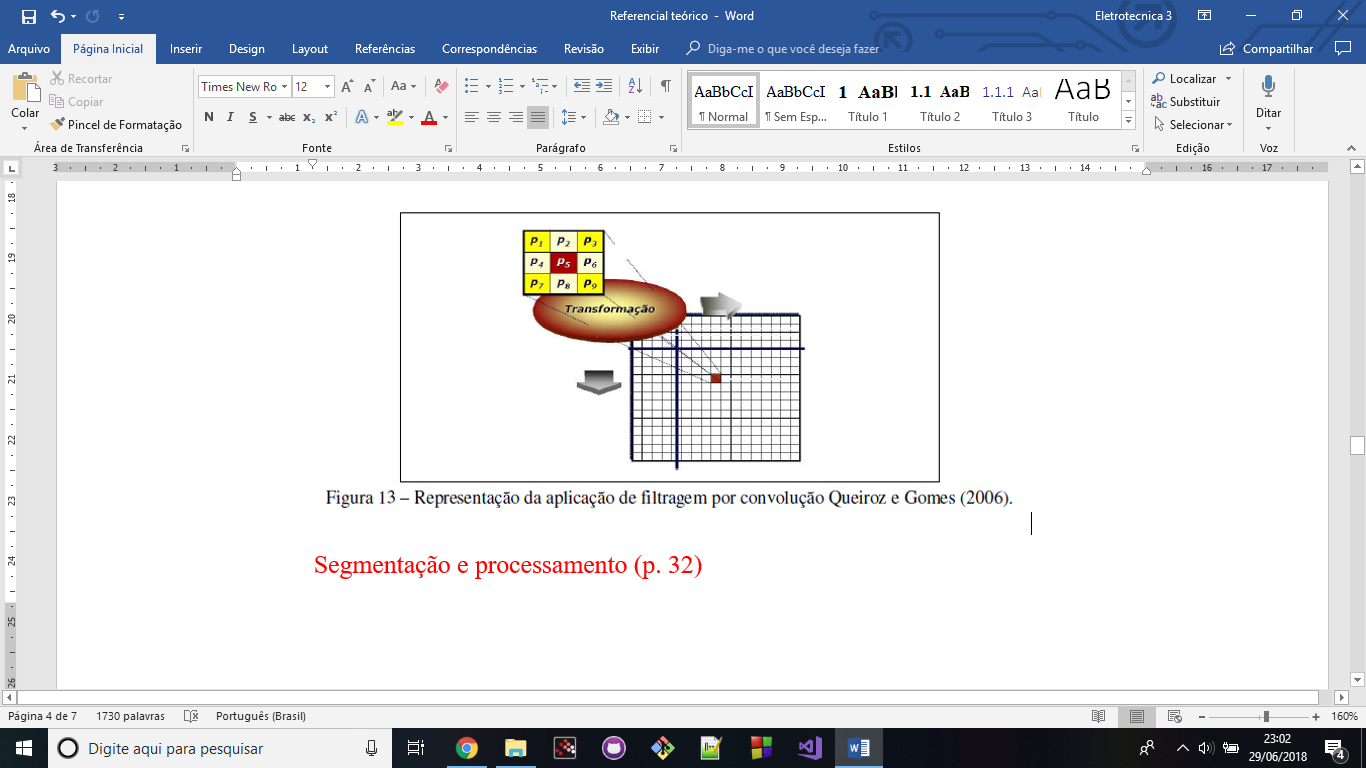
Para o retorno de uma imagem binária, representando as bordas de objetos, o método Canny realiza quatro tipos de operações sobre a fotografia original. Veja nos tópicos a seguir.

### Suavização Gaussiana

A primeira operação aplicada sobre a imagem é a filtragem conhecida como uma Suavização Gaussiana, que terá como objetivo diminuir ruídos sobre a imagem e facilitar a detecção de bordas. Por ruídos, entende-se como sendo frequências muito altas de variações de cores. A suavização se baseia em tirar a média ponderada da intensidade de cada pixel com os seus vizinhos. Diferente da transformação em escala de cinza que tirava a média ponderada da escala RGB do pixel. Os vizinhos mais próximos do pixel terão maior peso, enquanto os mais distantes terão menor relevância (gaussiana p 4).

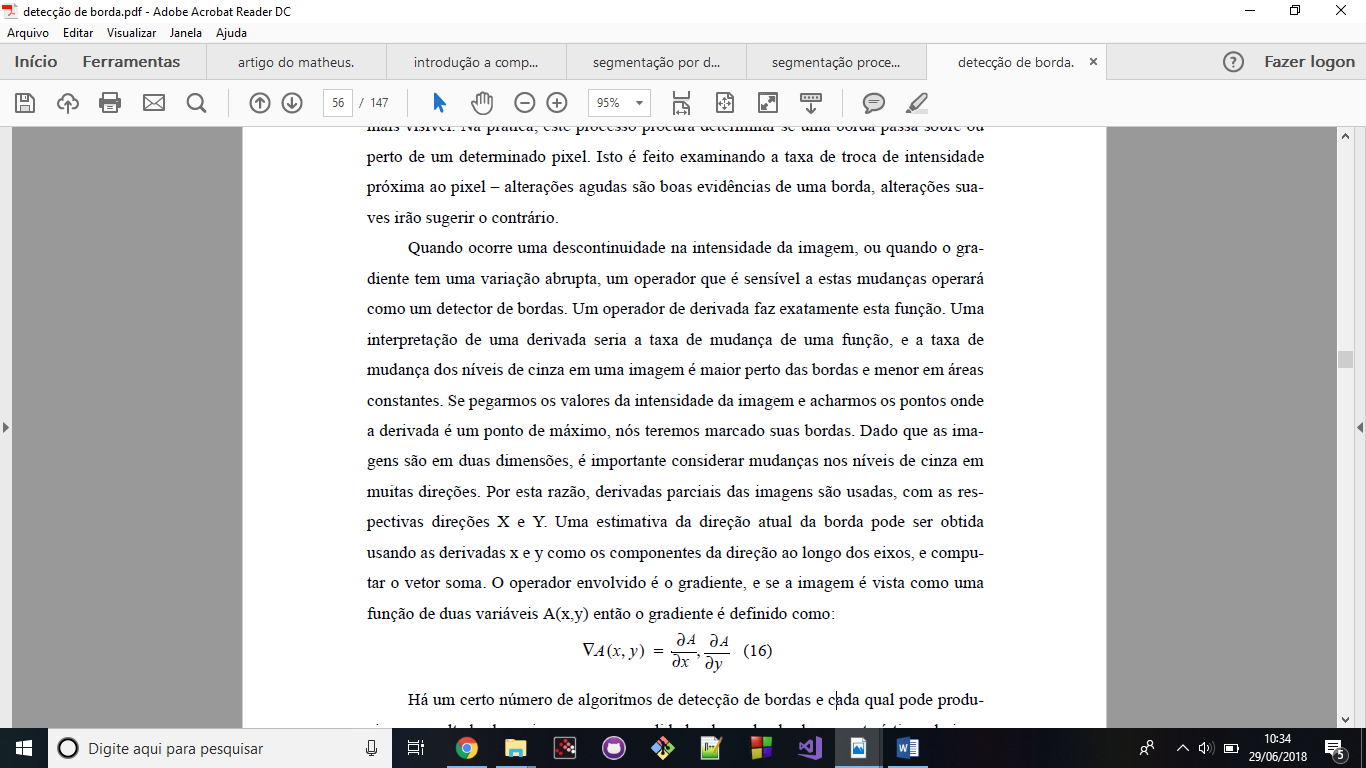
### Convolução

Na próxima etapa, aplica-se a ideia de convolução sobre a imagem, de forma que uma matriz de operação 3x3 filtra pixel a pixel e encontra uma relação entre o elemento e a intensidade de variação dos seus oito vizinhos (artigo do Matheus p8), levando em consideração pesos para cada um desses, o que equivale a primeira derivada.



Segmentação e processamento (p. 32)

As imagens são formadas por duas dimensões, dessa forma para melhorar o processo de encontrar a variação da intensidade de cor se aplicar a convolução nos dois eixos, x e y, simulando uma derivada parcial sobre a imagem (detecção de borda, 56). O representante dessas derivadas será denominado gradiente, calculado como sendo o vetor resultante das derivadas em um certo pixel. O gradiente pode ser dado por:



Essa formula apresenta o gradiente no pixel A(x,y) com componente x igual a derivada do pixel em função do eixo x e y igual a derivada do mesmo pixel em função de y. O tamanho do gradiente identifica a intensidade da derivada sobre um elemento na imagem, além disso, o seu sentido e direção aponta para onde há a maior variação de cor (introdução p 21).

G = (Gx^2 + Gy^2)^1/2

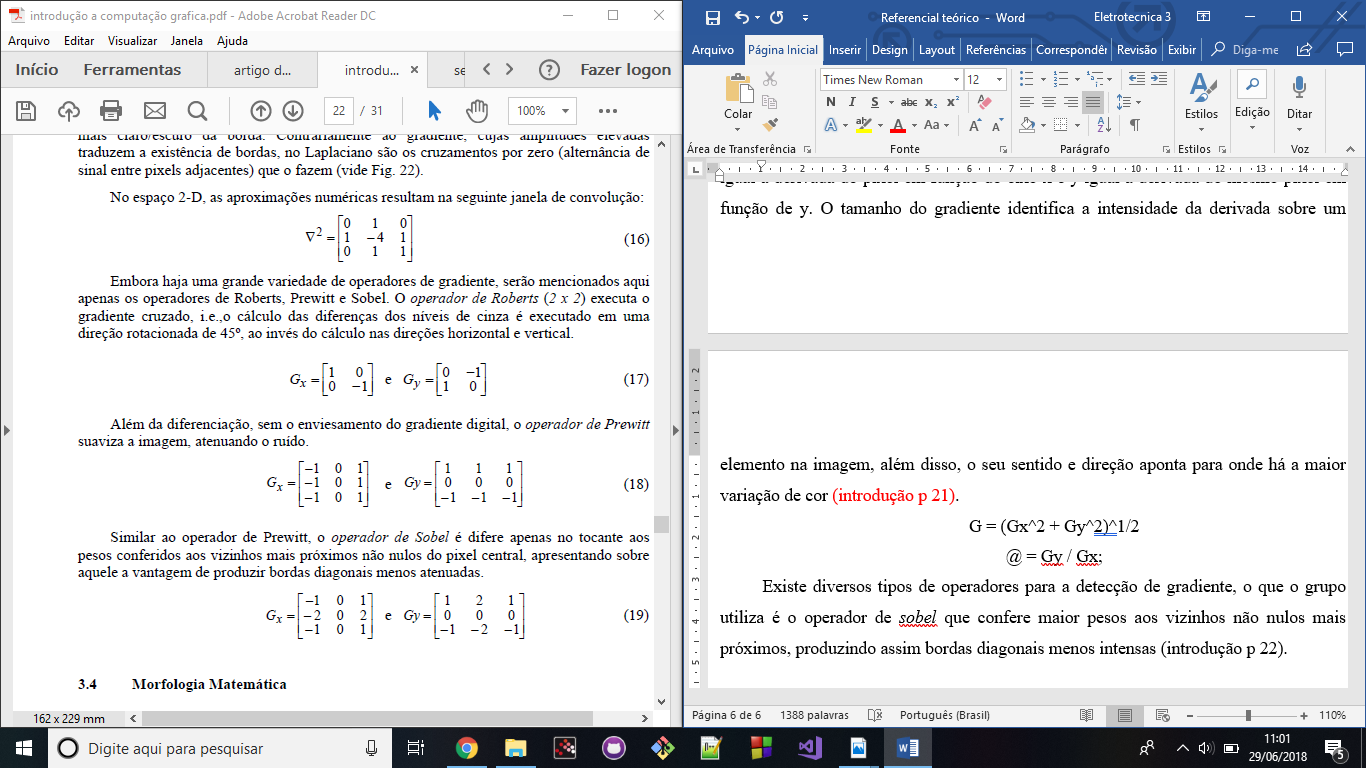
@ = Gy / Gx;

É importante ressaltar que como tratamos cada pixel da imagem como um elemento de uma matriz, o ângulo do gradiente é limitado em 0, 45, 90 e 135, considerando apenas o primeiro e segundo quadrante. Dessa forma, quando @ resulta em um ângulo diferente desses há um arredondamento.



Fonte (artigo do Matheus)

Existe diversos tipos de operadores para a detecção de gradiente, o que o grupo utiliza é o operador de *sobel* que confere maior pesos aos vizinhos não nulos mais próximos, produzindo assim bordas diagonais menos intensas (introdução p 22).



Fonte (introdução, p 21)

### Supressão do não máximo

Após aplicar a convolução em toda a imagem se inicia o processo de filtrar os pontos que serão bordas. Inicialmente é feito supressão do não máximo, que zera todos os valores não máximos na direção do gradiente. Relembrando que os gradientes máximos serão os pontos onde houve maior variação na intensidade da cor em uma imagem. Essa supressão possibilita que a borda seja representada apenas por um único pixel.

### Limirialização

Por fim, há a limirialização, que segundo segmentação por detecção de borda (p . 35) são parâmetros que delimitam a atuação de procedimentos sobre a imagem. No contexto do grupo a limirialização define um intervalo de valores que são considerados como bordas na imagem.

Entretanto, um dos principais problemas é definir os valores que serviram de parâmetros, uma vez que determinadas variáveis podem interferir na qualidade da borda, como por exemplo, iluminação, qualidade da imagem e a distância do objeto até a câmera. Isto significa que, quando um mesmo sistema é colocado em ambientes diferentes a limirialização tende a não fornecer resultados similares (segmentação por detecção de borda p 25).

Após definir o intervalo acontece o processo conhecido com binarização da imagem, definido por segmentação processamento (p 29). Nessa etapa que há a separação do plano de fundo e do objeto, de forma que todos os valores diferentes de zero depois do processamento são considerados como borda e recebem o valor *true (1)*, e o restante *false (0)*.

Segundo Grando (2005) segmentação processamento (p 29) matematicamente a detecção de borda pode ser definida como

G(x,y) = {1 se f(x,y) > T

{ 0 se f(x,y)<= T

Onde f(x,y) é o pixel na posição x e y da imagem original, T é o parâmetro e G(x,y) é o pixel da imagem binária.

## Biblioteca para o processamento de imagem: Opencv

O OpenCV (Biblioteca de visão computacional de código aberto) pode ser utilizada em interfaces C++,Phyton e Java e é disponível para uso em Windows, Linux, Mac OS, iOS e Android. Sendo projetada para eficiência computacional, seu foco é trabalhar com aplicações em tempo real.

Segundo Marengoni e Stringhini (2009, p127):

“A biblioteca OpenCV foi desenvolvida pela Intel e possui mais de 500 funções [4].Foi idealizada com o objetivo de tornar a visão computacional acessível a usuários e programadores em áreas tais como a interação humano-computador em tempo real e a robótica. A biblioteca está disponível com o código fonte e os executáveis (binários) otimizados para os processadores Intel. Um programa OpenCV, ao ser executado, invoca automaticamente uma DLL (Dynamic Linked Library) que detecta o tipo de processador e carrega, por sua vez, a DLL otimizada para este. Juntamente com o pacote OpenCV é oferecida a biblioteca IPL (Image Processing Library), da qual a OpenCV depende parcialmente, além de documentação e um conjunto de códigos exemplos.

Existem cinco grupos de funções para essa biblioteca, sendo eles: Processamento de imagens; Análise estrutural; Análise de movimento e rastreamento de objetos; Reconhecimento de padrões e Calibração de câmera e reconstrução 3D. (MARENGONI;STRINGHINI ,2009, p127).

Como a Open CV trabalha com a visão computacional, têm-se diversas maneiras de se trabalhar com imagens. Em alguns casos analisa-se um quadro fixo carregado de algum lugar. Em outros trabalha-se com streaming(modo ininterrupto de captar imagens de uma câmera) com dados em tempo real de algum dispositivo ou câmera. Na biblioteca mais especificamente, na parte HighGUI,é fornecida uma solução para essa situação. (BRADSKY;KAEHLER,2008,p 26).

O HighGUI pode ser dividido em três partes: hardware, sistema de arquivo e GUI. A parte do hardware está relacionada com o funcionamento das câmeras. O HighGUI permite uma maneira fácil de consultar recuperar a imagem mais recente da câmera. A parte do sistema de arquivo carrega e salva imagens. Uma característica da biblioteca é que se podem ler vídeos usando os mesmos métodos que se utiliza para ler uma câmera. Portanto se abstrai do dispositivo utilizado e continua-se escrevendo um código interessante. A parte GUI é o sistema de janelas que fornece funções que permitem abrir uma janela e abrir uma imagem dentro dessa janela. Com isso é permitido responder a comandos com o mouse e teclado. (BRADSKY;KAEHLER,2008,p 90).

# Metodologia

# Apresentação dos resultados

# Considerações finais

As imagens transmitem diversas informações sobre o mundo, como fatores históricos e culturais de uma determinada sociedade, noções biológicas de um ser vivo utilizando o raio – x, ou ainda informações da estrutura física de determinados objetos. O intuito desse trabalho era entender como a derivada poderia auxiliar no estudo dessa última informação, esclarecendo para os alunos do primeiro ano a aplicabilidade do Cálculo na Engenharia de Computação.

A área de foco foi o processamento de imagem, mas especificamente a detecção de bordas utilizando o método de Canny. Essa área possuía vantagens que possibilitava visualizar a derivada e exigia pouco conhecimento para entender, de forma básica, como se produzia uma imagem de bordas como saída do processamento. Além disso, outra razão pela escolha dessa área, foi a possibilidade de visualizar o processamento de imagens capturadas pela Webcam, o que possibilitariam aos alunos avaliar, por meio de algoritmos, o ambiente ao seu redor em tempo real. A análise proposta por esse trabalho foi utilizar o método Canny para medir objetos.

A principal dificuldade encontrada foi conseguir filtrar as informações necessárias para se ter uma base do processamento sem que fosse exigido muito conhecimento. Além disso, o grupo entrou em contato com dificuldades existentes na área de computação visual, como: ruídos na imagem, bordas não desejadas, relação entre o tamanho do objeto e sua distância da câmera e etc.

Portanto, por meio desse trabalho foi possível constatar que o processamento de imagem é uma área promissora que integra os conceitos do Cálculo da derivada e a área da computação. Como estudos futuros, o grupo pretende aperfeiçoar as técnicas de processamento utilizadas nesse trabalho, embarcando conceitos de outros períodos, além do primeiro e segundo. As melhorias que o grupo almeja são (1) conseguir fazer medições podendo varia a rotação do corpo e a distância desse até a câmera sem a necessidade de configurar uma nova relação; (2) automatizar a limirialização de bordas para cada ambiente; por fim (3) otimizar as estruturas do algoritmo.

# Referencial bibliográficos