

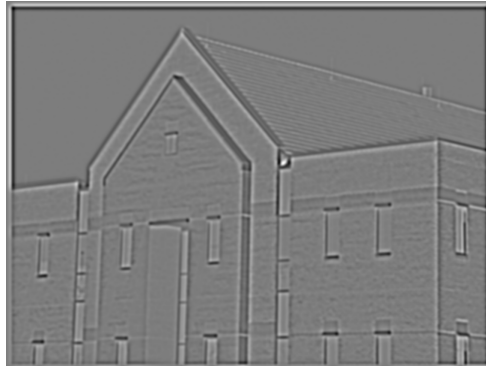
PROJETO 6 – Segmentação

Uma das primeiras tentativas bem-sucedidas de incorporar análises mais sofisticadas no processo de busca das bordas é atribuída a Marr e Hildreth (1980). O algoritmo proposto pode ser resumido da seguinte forma:

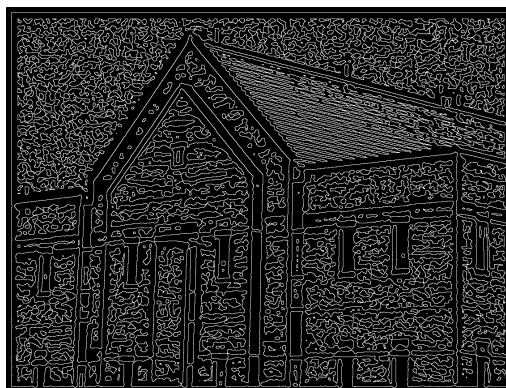
1. Filtrar a imagem de entrada com um filtro passa-baixas gaussiano.



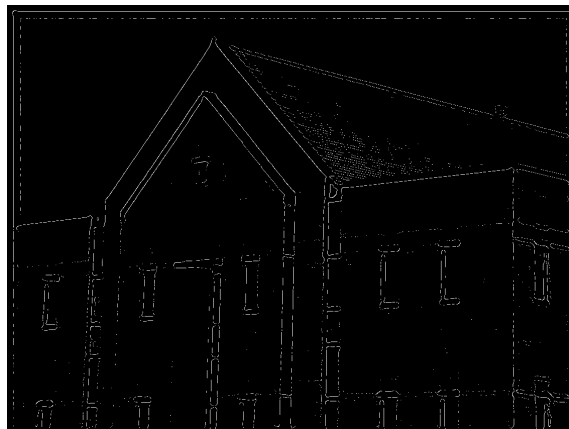
2. Calcular o laplaciano da imagem resultante da Etapa 1.



3. Encontrar cruzamentos por zero.



4. Realizar algum tipo de limiarização de forma a reduzir o efeito "espaguete"



O código abaixo realiza as três primeira etapas do algoritmo. Baseando-se na descrição do detector de bordas de Marr-Hildreth que se encontra no livro texto, implemente a etapa que falta.

```
clear all
close all
clc

I = imread('building_original.tif');
imshow(I);

% Filtro Gaussiano
hg = fspecial('gaussian', 25, 4);
Ig = conv2(double(I),double(hg));
figure;
imshow(Ig, []);

% Filtro Laplaciado
hl = [1 1 1; 1 -8 1; 1 1 1];
Ilog = conv2(double(Ig), double(hl));
figure;
imshow(Ilog, [])

% Calcula transições por zero

% Pega o sinal de cada elemento da matriz
sinais = sign(Ilog);
[h, w] = size(sinais);

% Verifica se os sinais de pixels opostos em cada direção
% têm sinais diferentes. Se tiverem, o valor central da região
% 3x3 será 0.

t1 = filter2([1 0 0; 0 0 0; 0 0 1],sinais);
t2 = filter2([0 1 0; 0 0 0; 0 1 0],sinais);
t3 = filter2([0 0 1; 0 0 0; 1 0 0],sinais);
t4 = filter2([0 0 0; 1 0 1; 0 0 0],sinais);

% Matriz que tem todas as transições por zero.
y = t1==0 | t2==0 | t3==0 | t4 == 0;

% Imagem Log mascarada pelo cruzamento em zero
IlogM = Ilog.*y;
% Pegando apenas os valores positivos (maiores que zero)
limiar = 0;
final = uint8(255*(IlogM>limiar));
figure
imshow(final)
```