

UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE COMPUTAÇÃO



MARCOS DA MATA SOUSA (221519)

Relatório - Trabalho 2

CAMPINAS

2022

1. Introdução

O presente relatório apresenta métodos de limiarização global e local. Métodos de limiarização global determinam um único valor de limiar para toda imagem. Este relatório, portanto, apresenta, principalmente, métodos de limiarização adaptativa local, ou seja, métodos que calculam um limiar para cada pixel da imagem com base na informação contida numa vizinhança de pixels. Em qualquer um dos casos, um pixel $p(x, y)$ é classificado como objeto (preto) se este possuir um valor maior que o limiar, e branco, caso contrário.

2. O problema dos cantos da imagem

A linha de código utilizada para extrair a vizinhança de centrada em um pixel $p(i, j)$ foi `vizinhos = img[i - n: i + n + 1, j - n: j + n + 1]` (um pouco melhor explicado na seção 3). Porém, foi possível observar pela depuração de código que, se o valor de j for igual a zero, então a resposta da vizinhança será vazia. Isso acontece pois não é possível começar a extração da vizinhança a partir de uma linha ou coluna (i, j) com valores menores do que 0.

A ideia utilizada para resolver este problema, foi, basicamente, adicionar paddings de cor preta de tamanho N aos cantos da imagem. O algoritmo para essa adição pode ser encontrado na função `addBordas(img, n)`. É possível observar, porém, que as imagens resultantes possuíram uma pequena alteração nos cantos, devido a adição dos paddings.

3. Extração de vizinhança

Após a adição dos paddings, explicada na seção 2, inicia-se a extração das vizinhanças, que, para todos os algoritmos de limiarização aqui presentes, acontecerá da mesma forma (note que o valor de N é alterável, pois este é um parâmetro das funções).

A extração da vizinhança funciona, basicamente, através de dois loops encadeados que iteram nas variáveis i e j , respectivamente. Portanto, para cada pixel $p(i, j)$, utiliza-se o fatiamento de matrizes para a extração. Como visto na seção 2, a linha de código referente a este procedimento é apresentada abaixo:

```
vizinhos = img[i - n: i + n + 1, j - n: j + n + 1]
```

Na linha de código apresentada, *vizinhos* é a variável na qual será armazenada a vizinhança do pixel $p(i, j)$, *img* é a imagem original e n é o raio de cobertura centrada em i, j . Por exemplo, se n for igual a 1, teremos uma matriz 3x3,

4. Binarização de imagem

Após o processo de extração dos valores de limiar (processos que serão mostrados mais adiante), é possível binarizar imagens através destes valores. Para tal, utilizou-se do

método *where*, da biblioteca *NumPy*, tendo a condição de que, se um pixel $p(i, j)$ da imagem original for maior que o valor de limiar $l(i, j)$, então este pixel assume o valor 0. Caso contrário, o pixel assume valor 255.

5. Imagens utilizadas

As imagens apresentadas nas próximas seções adiante, serão resultados da binarização baseada em diferentes métodos de limiarização das imagens abaixo, todas as imagens originais encontram-se no diretório em formato *pgm*. Será possível observar como alguns métodos não se aplicam muito bem em determinadas imagens, provavelmente pela sua configuração de intensidade de nível de cinza e/ou diferentes configurações de iluminação.

As imagens originais podem ser encontradas no diretório, bem como os resultados de processamento das imagens, depois de sua execução.



Figura 1: Retina

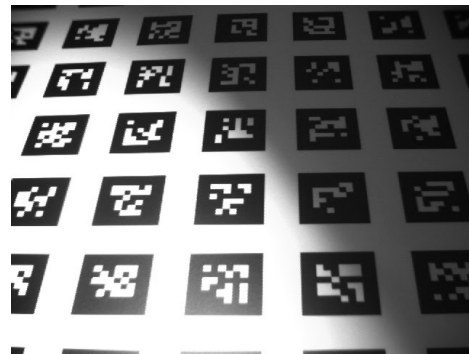


Figura 2: Fiducial

Abaixo encontram-se os histogramas para cada imagem de entrada:

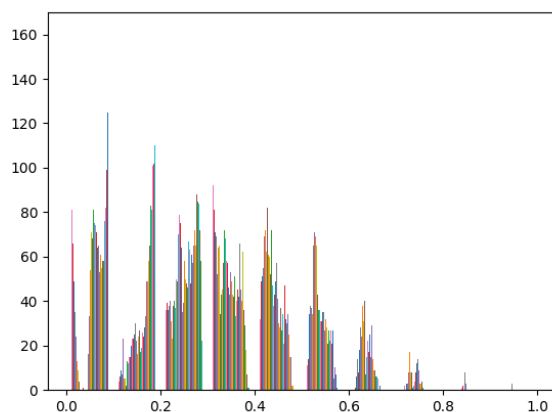


Figura 3: Histograma referente à Retina

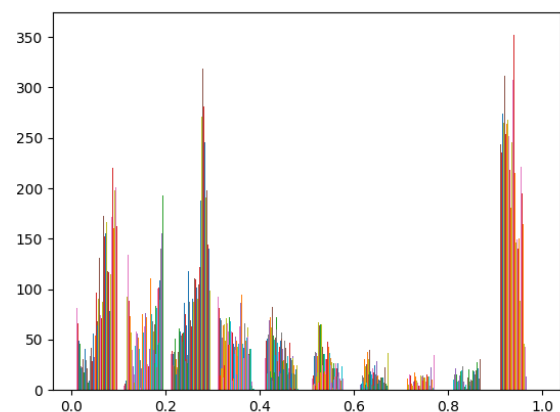


Figura 4: Histograma referente à Fiducial

6. Método global

O método global funciona com a simples ideia de que, se um pixel $p(i, j)$ possuir nível de intensidade de cinza maior que 128 (metade do intervalo $[0, 255]$), então este é classificado como objeto, caso contrário, é considerado branco.

O método é facilmente aplicado com o auxílio do método *where* da biblioteca *NumPy*, que estabelece o valor de cada pixel de uma imagem de entrada para 0, se este possuir nível de intensidade maior que 128, e para 255, caso contrário. O resultado obtido é apresentado abaixo.



Figura 5: Retina, método global

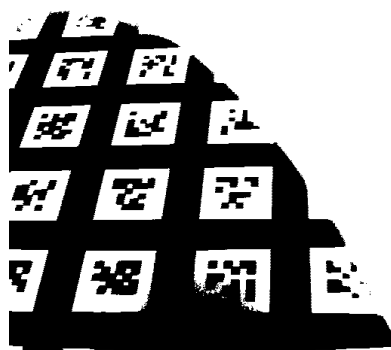


Figura 6: Fiducial método global

O resultado do método é bastante sólido para retina, podemos ver os “ramos” da retina com bastante clareza, porém, encontram-se alguns ruídos, principalmente na parte superior da imagem.

Já para Fiducial, o resultado não mostrou-se muito favorável, uma vez que apenas a parte iluminada da imagem original mostrou resultado sólido, enquanto sua outra metade foi tomada por branco.

7. Método de Bernsen

O método de Bernsen funciona com a simples premissa de determinar o limiar de um pixel $p(i, j)$ como a média do máximo e mínimo de uma vizinhança $N \times N$ centrada em i, j . Abaixo temos o resultado do método aplicado à retina e fiducial.

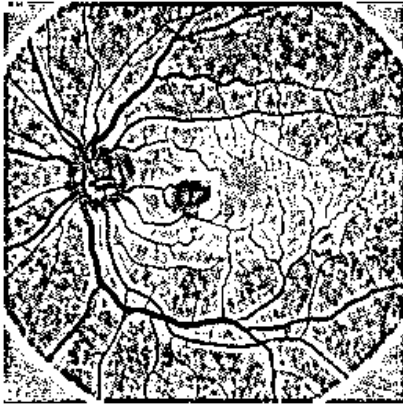


Figura 7: Retina, Bernsen com vizinhança 3x3

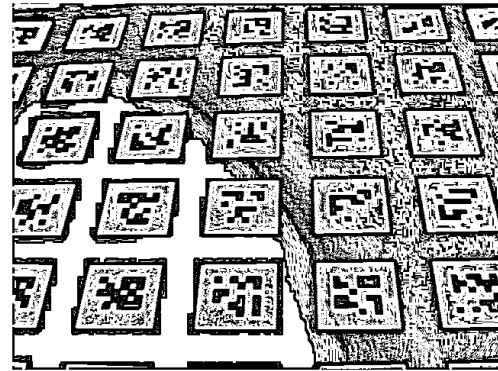


Figura 8: Fiducial, Bernsen com vizinhança 3x3

É possível observar, como previsto anteriormente, que a imagens possuem um pequena borda de tamanho 3x3 em cada um de seus cantos, devido ao padding adicionado na imagem original antes do processo de extração de vizinhança. Além disso, valores maiores do que 3, parecem engrossar os traços da imagem de maneira que não parecem fiéis à sua representação original.

Quanto ao resultado em si, a retina parece uma imagem razoável para propósitos médicos, por exemplo (não que eu tenha algum conhecimento médico). É possível observar, apesar das pequenas perdas de informação, a maior parte dos “ramos” presentes na retina, apesar da grande quantidade de ruídos.

Com relação à segunda imagem, fiducial, obteve resultado significativamente melhor do que o método global, uma vez que todos os quadrados agora são visíveis mesmo numa representação da imagem binarizada. No entanto, muitos ruídos ainda são visíveis na imagem

8. Método de Niblack

No método de Niblack, o limiar de um pixel $p(i, j)$ é determinado através da equação:

$$T(i, j) = \mu(i, j) + k\sigma(i, j)$$

onde k é usado para ajustar a fração da borda do objeto a ser considerada como parte do objeto. Usou-se o valor de $k = 1$ e o resultado é exibido abaixo.

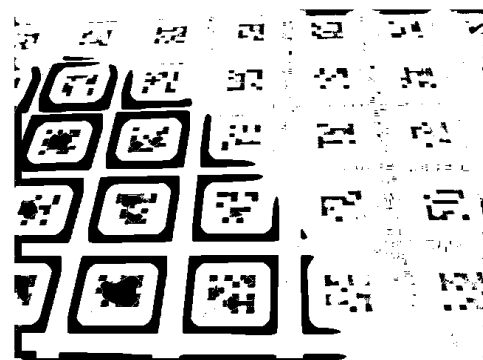
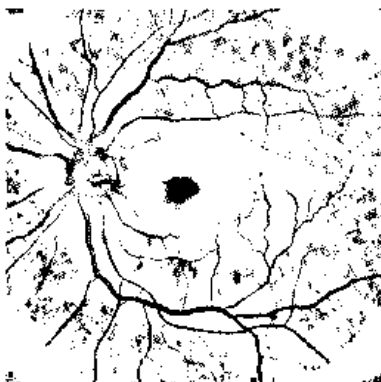


Figura 9: Retina, Niblack vizinhança 20x20

Figura 10: Fiducial, Niblack vizinhança 15x15

Como pode ser observado, o método de Niblack pareceu obter mais sucesso do que o método de Bernsen em mostrar com mais clareza os “ramos” da retina. Por outro lado, alguns ruídos, algo que se parecem com manchas, ainda preenchem a imagem resultante em praticamente toda sua extensão.

Já para Fiducial, o método de Bernsen obteve resultado vantajoso, visto que no método de Niblack, a metade direita não teve seus quadrados representados e, quando teve (na metade esquerda), o centro destes apresentaram certas deformidades. Ainda assim, o método de Bernsen exibe uma quantidade bem maior de ruído, enquanto a de Niblack não.

Para valores menores de N , como 1 ou 2 (Figura 7), o método mostrou-se pouco eficiente, visto que, na retina, exibia “ramos” suficientemente finos para serem confundidos em meio aos ruídos. Por outro lado, o Fiducial conseguiu mostrar um resultado razoável no lado direito da imagem, mas o esquerdo apresentou apenas ruídos. Dois exemplos podem ser vistos abaixo, com k também igual a 1. Esta situação já havia sido prevista, então o resultado ocorreu como esperado, pois necessitamos de uma vizinhança suficientemente pequena para preservar detalhes, mas suficientemente grande para suprimir ruídos (tanto para este, quanto outros métodos).

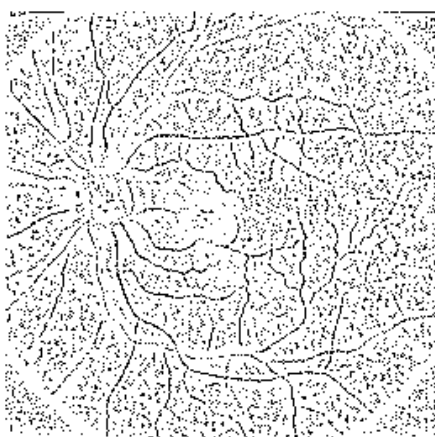


Figura 11: Retina, Niblack, vizinhança 2x2

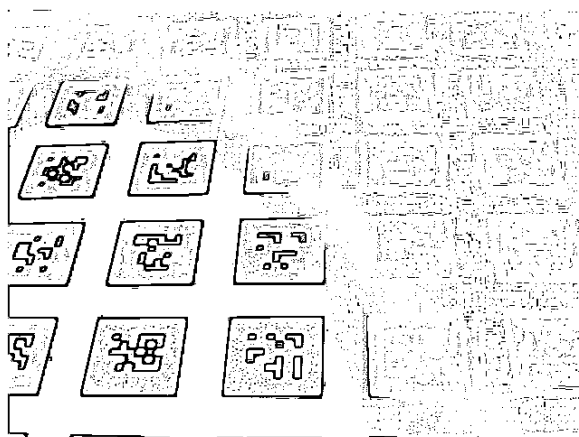


Figura 12: Fiducial, Niblack, vizinhança 2x2

9. Método de Sauvola e Pietaksen

Este método é apresentado como uma proposta de melhoria do método de Niblack, principalmente para imagens de documentação com má iluminação. O método para cálculo de limiar de um pixel $p(i, j)$ é dada pela equação:

$$T(i, j) = \mu(i, j) = \left[1 + k \left(\frac{\sigma(i, j)}{R} - 1 \right) \right]$$

Vale lembrar que os autores sugerem valores de 0.5 e 128, para k e R , respectivamente. Abaixo podemos ver os resultados do método:

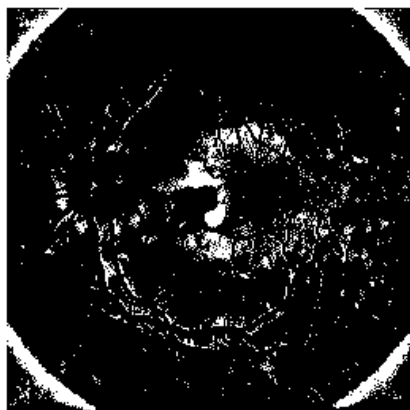


Figura 13: Retina, Método de Sauvola, vizinhança 10x10

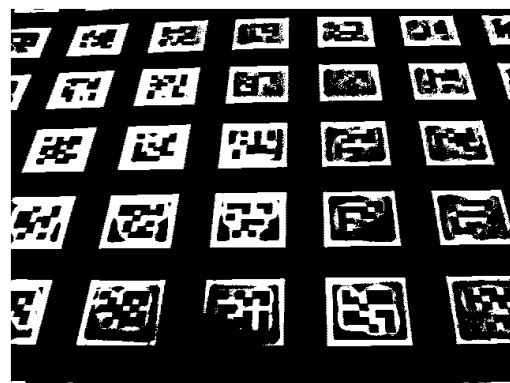


Figura 14: Fiducial, Método de Sauvola, 10x10

Este é um dos casos previstos na seção 5. Podemos observar que o resultado do método aplicado sobre a Retina não foi bom. Não é possível fazer quase nenhuma distinção sobre os “ramos” da retina. Por outro lado, vemos que na imagem Fiducial, o resultado é bastante sólido, o método foi capaz de oferecer uma boa distinção dos objetos presentes na imagem e realmente foi capaz de apresentar uma melhoria com relação ao método de Niblack.

10. Método de Phansalskar, More e Sabale

Este método é descrito como uma variação do método anterior para lidar com imagens de baixo contraste. O limiar de um pixel $p(i, j)$ é definido de acordo com a equação abaixo:

$$T = \mu(i, j)[1 + p * \exp(-q * \mu(i, j)) + k * (\frac{\sigma(i, j)}{R} - 1)]$$

Os autores deixam como sugestão os valores $k = 0.25$, $R = 0.5$, $p = 2$ e $q = 10$. É possível observar abaixo a aplicação do método sobre as duas imagens:

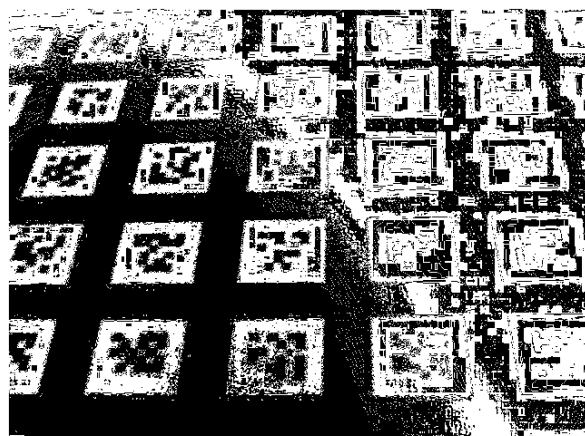
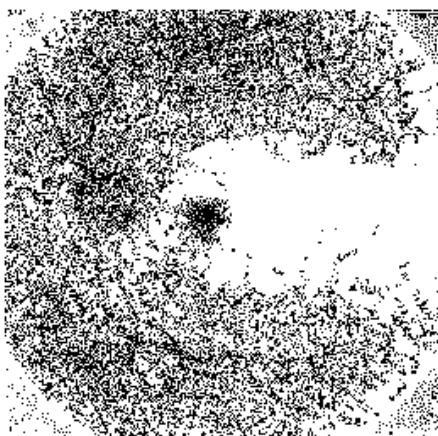


Figura 15: Retina, Phansalskar, 1x1

Figura 16: Fiducial, Phansalskar, 1x1

Referente ao resultado obtido para a retina, é praticamente certo dizer que foi o pior resultado que obtemos dentre todos os métodos. A imagem resultante teve boa parte de suas informações ocultadas, ao mesmo tempo em que as informações presentes são apresentadas por ruídos muito indistintos, tornando praticamente impossível a análise da imagem por um médico, por exemplo.

Para o fiducial, o método também não apresentou resultado muito satisfatório, embora pareça menos pior do que o resultado obtido na retina. Podemos observar que a metade esquerda da imagem obteve resultado melhor do que o lado direito, o que também pode nos levar a imaginar que o método se sai melhor em algumas configurações de intensidade de cinza e/ou iluminação

11. Método do Contraste

O método do contraste atribui o valor de pixel como fundo ou objeto, dependendo se o valor original do pixel está mais próximo do máximo ou do mínimo local da vizinhança, respectivamente. Como mostram os resultados abaixo, o método do contraste é análogo ao método de Bernsen.

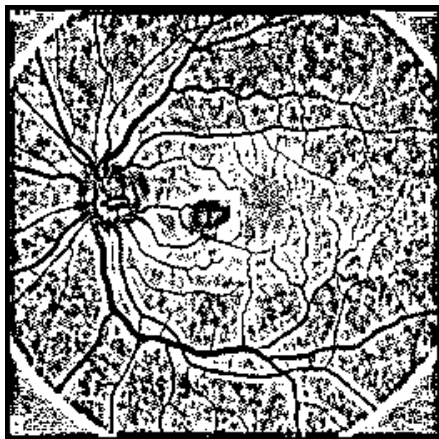


Figura 17: Retina, Método Contraste, 3x3

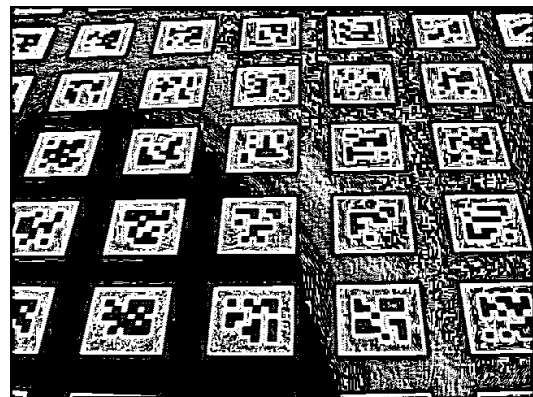


Figura 18: Fiducial, Método Contraste, 3x3

12. Método da Média

O método da média funciona de maneira simples, como sugere o seu nome. O limiar de um pixel $p(i, j)$ é determinado a partir da média aritmética de uma vizinhança centrada em i, j . Abaixo temos o resultado do método aplicado na Retina, com duas configurações diferentes de vizinhança.

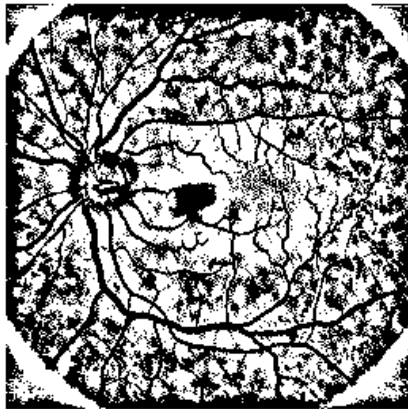


Figura 19: Retina, método Média, 10x10



Figura 20: Retina, método Média, 20x20

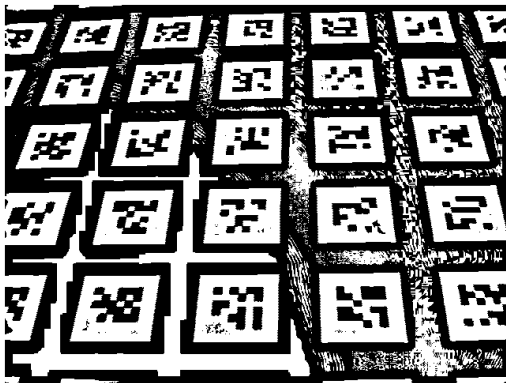


Figura 21: Fiducial, método Média, 10x10

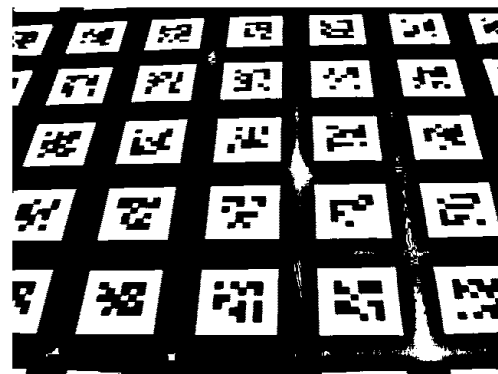


Figura 22: Fiducial, método Média, 20x20

É possível observar nos resultados acima que, na retina, uma vizinhança maior oferece uma quantidade maior de ruídos nas bordas da retina e, apesar de diminuir os ruídos na parte central, uma quantidade considerável ainda pode ser observada. Na vizinhança menor, porém, apesar da grande quantidade de ruídos, ainda parece ser possível distinguir a maior parte dos “ramos” da retina com certa facilidade.

Já para Fiducial, é possível observar uma definição muito maior e menos ruídos na vizinhança de 20x20, apesar de ainda apresentar pequenos ruídos brancos, mas ainda assim muito menos graves do que os ruídos observados com o método numa vizinhança de 10x10

13. Método da Mediana

No método da mediana, também usa-se uma abordagem simples: para cada um dos pixels $p(i, j)$, o limiar é obtido a partir da mediana dos valores da vizinhança $N \times N$. Abaixo vemos os resultados aplicados na Retina e Fiducial.

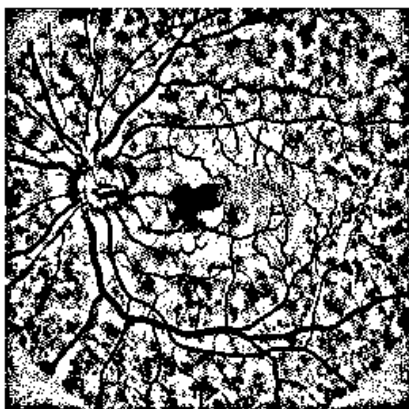


Figura 23: Retina, Método Mediana, 10x10

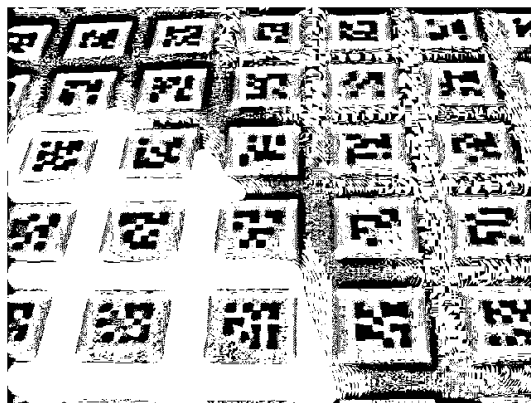


Figura 24: Retina, Método Mediana, 10x10

É observável que, para a retina, o método da mediana obteve resultado similar ao método da média, apresentando uma boa distinção dos “ramos”, mas exibindo uma quantidade bastante grande de ruídos. Já para fiducial, o resultado foi inferior. Ainda sem exibir uma definição razoavelmente boa, o método da mediana apresentou ainda mais ruídos e, mesmo com o aumento do tamanho da vizinhança, o problema não pôde ser resolvido.

Neste relatório vimos diferentes métodos para obtenção de limiares de imagens monocromáticas, bem como resultados das aplicações dos métodos em diferentes experimentos, variando-se, por exemplo, o tamanho de uma vizinhança entre duas imagens diferentes.