

Relatório de experimentos com filtragem

1. Especificações

ERICK ECKERMANN CARDOSO
GRR20186075

Terceiro trabalho de laboratório da disciplina de Processamento de imagens

Código escrito em Python3, utilizando as seguintes bibliotecas:

- Open-cv (cv2)
- sys
- numpy
- random

O código está contido inteiramente no arquivo filtro.py

Antes de executar, certifique que o arquivo da imagem esteja no mesmo diretório que o executável, pois o código lê a imagem a partir do diretório corrente.

Para compilar e executar o programa, basta utilizar o comando python3 com o nome do executável, seguido pela imagem original, nível de ruído, tipo de filtro e o nome da imagem de saída, da seguinte maneira:

```
$ python3 filtro.py <IMG ORIGINAL> <NIVEL RUÍDO> <FILTRO> <IMG SAÍDA>
```

ex: python3 filtro.py original.png 0.01 1 out.png

O programa criará uma imagem com o nome especificado, sendo essa a imagem com o filtro aplicado.

2. Resultado dos experimentos

Foram realizados diversos experimentos com os filtros implementados, diferenciando o nível de ruído das imagens geradas a partir da original.

Para comparação entre as imagens foi utilizada a métrica PSNR, implementada pelo Open-cv, evitando assim somente uma análise visual.

Os níveis de ruído utilizados foram: [0.01, 0.02, 0.05, 0.07, 0.1].

2.1. Ruído 0.01

Para esse nível de ruído o filtro por empilhamento foi o melhor, com uma taxa de em média 30.5 de PSNR. A partir de 15 imagens empilhadas não houve mais um crescimento significativo na taxa PSNR, se mantendo abaixo de 31 sempre.

Os filtros de média e mediana apresentaram resultados similares, onde quanto menor o tamanho da máscara, maior a semelhança com a imagem original. A média apresentou maior taxa de 26.5 com máscara de tamanho 3 e a mediana apresentou maior taxa de 27.3 com máscara também de tamanho 3.

2.2 Ruído 0.02

Para o ruído 0.02 os resultados foram similares, se mantendo o melhor filtro por empilhamento, com taxa PSNR em média de 29.2, não tendo mais acréscimos significativos a partir de 15 imagens empilhadas. O filtro da média se manteve em 24 com o menor tamanho de máscara possível, de 3 pixels, e o da mediana se comportando da mesma maneira teve taxa de similaridade em 27. Quanto maior o tamanho da máscara, menor a similaridade pois aumenta o efeito de blur.

2.3 Ruído 0.05

Para esse caso o filtro que obteve melhor desempenho foi a mediana, novamente com o menor tamanho de máscara, 3 pixels, obteve taxa de 26.9. Para a média, notamos uma diferença pois agora o melhor desempenho foi com tamanho de máscara 7, onde o resultado apresentou uma taxa PSNR de 23. Para o filtro de empilhamento, devido a alta quantidade de ruído, começa a obter menos desempenho, sendo o máximo alcançado de 25.5 com 20 imagens empilhadas, sem obter acréscimos significativos a partir disso.

2.4 Ruído 0.07

Média, máscara de 9 pixels com resultado de 22.07. Para a média, novamente com o menor tamanho de 3 pixel na máscara obteve resultado de 26.26, sendo o melhor filtro nesse caso novamente.

O filtro por empilhamento não obteve desempenho satisfatório, não conseguindo desempenho maior que 24 de taxa PSNR, com 20 imagens empilhadas.

2.5 Ruído 0.1

A média obteve melhor resultado com máscara de tamanho 13, com resultado de 21. A mediana continuou na liderança, mantendo o padrão de quanto maior o tamanho da máscara, melhor o resultado, apresentando taxa de semelhança de 25.14.

Por empilhamento obteve taxa de 22 com 20 imagens empilhadas, sem obter acréscimos significativos a partir disso.

3. Conclusão

Analisando o tipo de ruído salt and pepper percebe-se que o filtro da mediana é o que se obtém melhor filtragem para o caso de ruído geral, com a máscara de pixels tamanho 3. O filtro de empilhamento apresenta resultado visual satisfatório, e funciona muito bem para ruídos com menor nível de intensidade. Para todos os casos, o filtro sobre a média não é recomendado para o ruído salt and pepper.